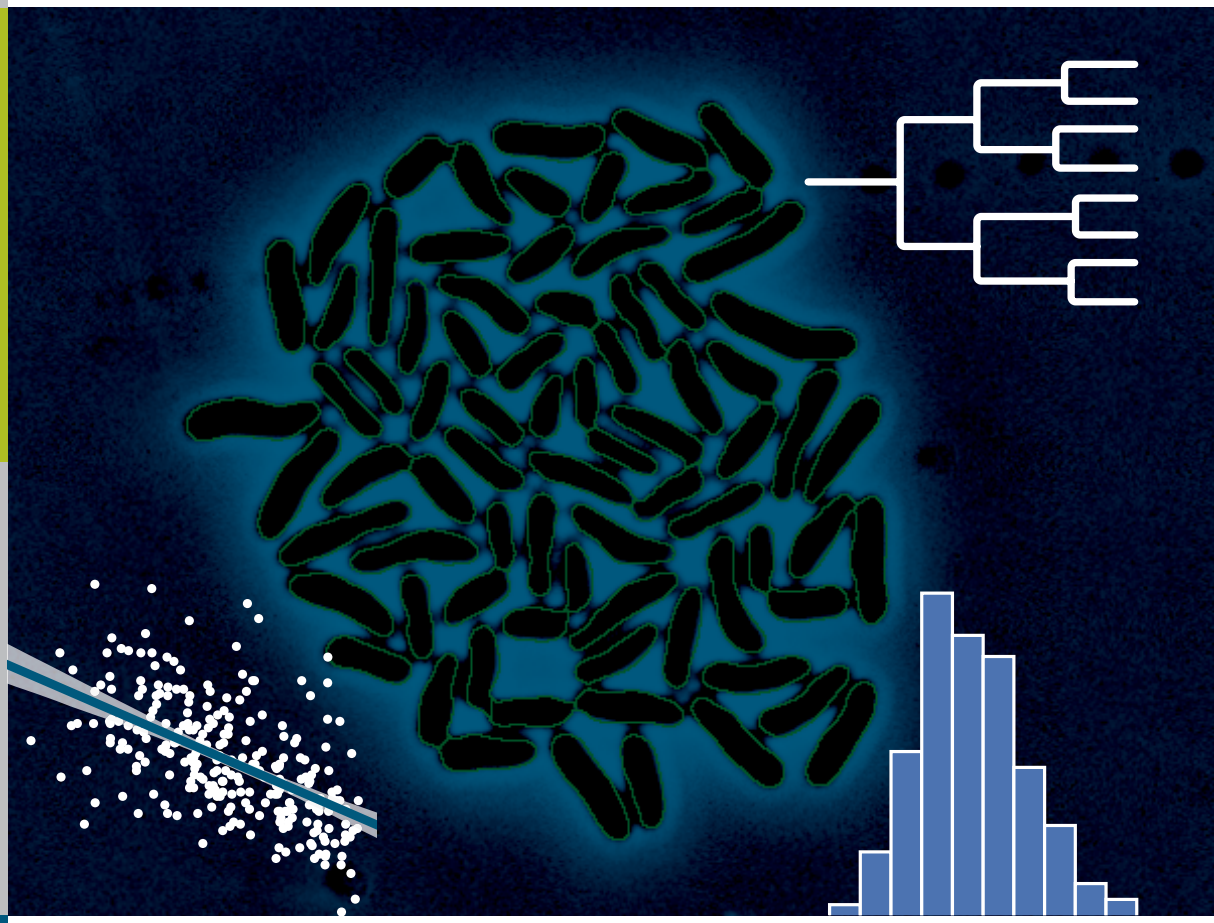
High-Throughput Live-Cell Imaging for Investigations of Cellular Heterogeneity in *Corynebacterium glutamicum*

Stefan Helfrich



Member of the Helmholtz Association

Schlüsseltechnologien / Key Technologies
Band / Volume 130
ISBN 978-3-95806-167-5

JÜLICH
FORSCHUNGSZENTRUM

Schriften des Forschungszentrums Jülich
Reihe Schlüsseltechnologien / Key Technologies Band / Volume 130

Forschungszentrum Jülich GmbH
Institute of Bio- and Geosciences
Biotechnology (IBG-1)

High-Throughput Live-Cell Imaging for Investigations of Cellular Heterogeneity in *Corynebacterium glutamicum*

Stefan Helfrich

Schriften des Forschungszentrums Jülich
Reihe Schlüsseltechnologien / Key Technologies Band / Volume 130

ISSN 1866-1807 ISBN 978-3-95806-167-5

Bibliographic information published by the Deutsche Nationalbibliothek.
The Deutsche Nationalbibliothek lists this publication in the Deutsche
Nationalbibliografie; detailed bibliographic data are available in the
Internet at http://dnb.d-nb.de.

Publisher and Distributor:	Forschungszentrum Jülich GmbH Zentralbibliothek 52425 Jülich Tel: +49 2461 61-5368 Fax: +49 2461 61-6103 Email: zb-publikation@fz-juelich.de www.fz-juelich.de/zb
Cover Design:	Grafische Medien, Forschungszentrum Jülich GmbH
Printer:	Grafische Medien, Forschungszentrum Jülich GmbH
Copyright:	Forschungszentrum Jülich 2016

Schriften des Forschungszentrums Jülich
Reihe Schlüsseltechnologien / Key Technologies, Band / Volume 130

D 82 (Diss., RWTH Aachen University, 2016)

ISSN 1866-1807
ISBN 978-3-95806-167-5

The complete volume is freely available on the Internet on the Jülicher Open Access Server (JuSER)
at www.fz-juelich.de/zb/openaccess.

Danksagung

Diese Dissertation wäre in ihrer jetzigen Form nicht ohne die Hilfe vieler Menschen möglich gewesen. Ich möchte diesen Personen an dieser Stelle meinen Dank aussprechen.

Mein besonderer Dank gilt meinem Doktorvater Prof. Dr. Wolfgang Wiechert, Leiter der Systembiotechnologie am Institut für Bio- und Geowissenschaften 1 am Forschungszentrum Jülich, für die Möglichkeit dieses Thema zu bearbeiten und für die Schaffung guter Arbeitsbedingungen am Institut. Außerdem möchte ich mich bei Prof. Dr. Björn Usadel als Zweitberichter dieser Arbeit bedanken.

Ich möchte Dr. Katharina Nöh als direkte Betreuerin dieser Arbeit für ihre anhaltende Unterstützung danken. Unsere Treffen und Diskussionen konnten mir immer wieder eine neue Perspektive aufzeigen, als ich glaubte in eine Sackgasse gelaufen zu sein. Des Weiteren haben ihre Kommentare und das kritische Hinterfragen maßgeblich zur der Qualität von Veröffentlichungen, Postern, Vorträgen, und schlussendlich dieser Arbeit beigetragen.

Für eine großartige, produktive Arbeitsumgebung möchte ich mich bei allen (ehemaligen) Mitgliedern der Arbeitsgruppe “Modellierung und Simulation” bedanken.

Darüber hinaus geht mein Dank an Juniorprof. Dr. Dietrich Kohlheyer und Juniorprof. Dr. Julia Frunzke. Die Diskussionen zu Themen der Mikrofluidik sowie der Mikrobiologie und Biotechnologie haben erheblich zum Erfolg dieser Arbeit und zur Erweiterung meines wissenschaftlichen Horizontes beigetragen. Die Kooperation mit ihnen und allen (ehemaligen) Mitgliedern ihrer Arbeitsgruppen hat zu einer erfolgreichen ersten Förderperiode des DFG Schwerpunktprogramms “Phänotypische Heterogenität und Soziobiologie in bakteriellen Populationen” (SPP1617), sowie zu einer Verlängerung des Programms geführt. Dankend möchte ich an dieser Stelle Christina Krämer, Antonia Heyer, Eugen Pfeifer und Arun Nanda erwähnen, mit denen mir die reibungslose Zusammenarbeit im Rahmen des SPP1617 immer große Freude bereitet hat.

Von unverzichtbarer Bedeutung für das Gelingen dieser Arbeit war die enge und produktive Zusammenarbeit mit Dr. Alexander Grünberger. Sein experimentelles sowie

biologisches Wissen hat in, teils kontroversen, Diskussionen zu immer neuen Ideen geführt und die Einzelzellanalyse und -auswertung am Institut voran getrieben. Ohne unsere intensive Zusammenarbeit und seine Unterstützung wäre die Bildanalyse heute vielleicht noch kein fester Bestandteil der Einzellzellanalyse am IBG-1.

Großer Dank gilt auch den von mir betreuten Studenten Charaf E. Azzouzi (Praxissemester, Master), Karin Bokelmann (Master) und Axel Theorell (Master) die die Bildanalyse am IBG-1 durch ihre eigenen Ansichten und Diskussionen vorangetrieben haben. Zusätzlich möchte ich mich bei Johannes Seiffarth bedanken, der mich als Werksstudent tatkräftig unterstützt hat.

Die exzellenten Rahmenbedingungen für das Gelingen der vorliegenden Arbeit verdanke ich der technischen und administrativen Infrastruktur des IBG-1. Bedanken möchte ich mich deshalb bei Marianne Hess, Dr. Iris Eggeling und allen Mitarbeitern der Infrastruktur für die Unterstützung bei diversen technischen und administrativen Angelegenheiten. Besonderer Dank geht hier an Andreas Franz, Horst Kiehl und das restliche Team der IT Abteilung, die mir bei der Umsetzung diverser technischer Ideen beratend zur Seite standen und sich einbrachten.

Dank gilt auch Elisabeth Zelle, Sebastian Niedenführ und Alexander Grünberger für das Korrekturlesen dieser Arbeit.

Für die rege Teilnahme an interessanten Gesprächsrunden möchte ich der “Bildanalyserunde” danken. Obwohl wir unterschiedliche biologischen Systeme und damit auch Bildinhalte bearbeiten, tut es doch gut zu wissen, dass man mit diversen Herausforderungen nicht alleine ist.

Danken möchte ich außerdem meinen Eltern, Georg und Elisabeth, für ihre fortwährende Unterstützung während meines Studiums und vor allem dieser Promotion.

Abschließend möchten ich mich bei Verena bedanken, die mich während dieser Promotion immer unterstützt und ohne Murren ertragen hat.

Kurzfassung

In den vergangenen 15 Jahren wurden signifikante Unterschiede zwischen einzelnen Zellen im Bezug auf Wachstum, Stressresistenz und andere zelluläre Merkmale in monoklonalen Bakterienpopulationen beobachtet [1]. Fortschritte in Mikrofluidik und Mikroskopie ermöglichen es die Entwicklung einzelner Zellen mit bisher nicht gekannter räumlicher und zeitlicher Auflösung zu untersuchen. In Verbindung mit optimierten, mikrofluidischen Labs-on-a-Chip [2], können hunderte Bakterienpopulationen unter kontrollierten Umweltbedingungen kultiviert und Bildsequenzen aufgezeichnet werden.

Mit der Möglichkeit große Datenmengen automatisiert zu erzeugen, ist die Auswertung der Bilddaten zu dem entscheidenden Schritt für die Erhebung von quantitativen, zeitaufgelösten Informationen geworden. Daher wurde eine erweiterbare Bildanalyse-Pipeline für die Auswertung von Bildsequenzen des biotechnologischen Modellorganismus *Corynebacterium glutamicum* entwickelt. Die Pipeline ist für die Identifizierung von Zellen in gedrängten Umgebungen optimiert, sowie für das Tracking von Zellen bei großer räumlicher Verschiebung. Zusätzlich können eine Vielzahl von Zelleigenschaften, wie zum Beispiel morphologische Parameter und Fluoreszenzintensitäten, extrahiert werden.

Die Bildanalysepipeline ist als Plugin für die ImageJ(2) Plattform implementiert worden. ImageJ(2) stellt fortschrittliche Datenstrukturen zu Verfügung und ermöglicht das Steuern von Abläufen mit Hilfe einer graphischen Benutzeroberfläche. Die zugrundeliegende Service-Architektur fördert die Erweiterbarkeit und Flexibilität von Modulen, so dass diese auch in anderen Plugins verwendet werden können.

Mit einer Kombination aus Mikrofluidik, Live-Cell Imaging Verfahren und Bildanalysetechniken ist man in der Lage Heterogenität in mikrobiellen Populationen auch bei niedriger zeitlicher Auflösung zu quantifizieren. Während die Analyseplattform für eine Vielzahl von Untersuchungen angewendet wurde, liegt der Fokus in dieser Arbeit auf zwei Anwendungsfeldern: mikrobielles Wachstum und Morphologie sowie Prophageninduktion in *C. glutamicum*. Im Rahmen der Wachstumsstudien wird die Übertragbarkeit von etablierten Quantifizierungsmethoden auf Einzelzelldaten untersucht. Eine zweite Anwendung auf diesem Gebiet überträgt die gewonnenen Erkenntnisse auf eine Screening-Studie von *C. glutamicum*, in welcher der Einfluss der Medienkomposition auf das Wachstum und die morphologischen Parameter untersucht wird.

In einem zweiten Anwendungsfeld wird eine Analyse der mikrobiellen Stressantwort und die Induktion eines Prophagen in *C. glutamicum* beleuchtet. Zu diesem Zweck ist ein Dual-Reporter-Stamm (d.h. ein Stamm mit beiden Reportern für SOS-Antwort und Prophagen-Induktion) in mikrofluidischen Labs-on-a-Chip kultiviert und mittels Fluoreszenzmikroskopie analysiert worden. Aus den zeitaufgelösten Reporterdaten wurde ein Zellzustandsmodell abgeleitet, welches für die Populationsmodellierung von *C. glutamicum* verwendet wurde.

Abstract

Significant cell-to-cell variation with respect to growth, stress resistance, and other cellular traits are observed in clonal microbial populations [1]. Advances in lab-on-a-chip research and time-lapse microscopy have recently extended the experimental capabilities to observe the development of individual cells with unprecedented spatial and temporal resolution. In combination with appropriate cultivation devices, e.g., custom microfluidic lab-on-a-chip devices [2], image sequences are acquired for hundreds of developing populations in parallel under controlled environmental conditions.

With the possibility to generate such large-scale datasets, the role of image analysis has become a crucial step for the elicitation of quantitative, time-resolved information for direct interpretation as well as modeling purposes. We have developed an extensible image analysis pipeline for the evaluation of time-lapse videos of the industrially competitive amino-acid producer *Corynebacterium glutamicum*. The pipeline has been optimized for the identification of cells in crowded environments, tracking of cells with large spatial displacements, and the extraction of a multitude of cellular characteristics, for instance, cell morphology and fluorescence reporter intensities.

The presented pipeline is implemented as a plugin for the well established ImageJ(2) platform. The platform provides advanced data structures and allows for visual controls of workflow composition and parameters. The underlying service architecture promotes extensibility of modules and flexibility to use implementations in alternative contexts.

The combination of microfluidic system, live-cell imaging setup, and image analysis techniques is capable to address challenges of population heterogeneity in microbial populations even at low temporal resolution. While the analysis platform has been applied for a variety of studies, applications from two fields are highlighted in this thesis.

First, investigations of microbial growth and morphology of *C. glutamicum*. Here, the applicability of growth quantification methods from bulk experiments to single-cell data are investigated. A second application transfers this knowledge to a profiling study of *C. glutamicum* in which the influence of medium composition (i.e., carbon sources) on growth and morphology parameters is analyzed.

Furthermore, an analysis of the microbial SOS response and the induction of a prophage in *C. glutamicum* is presented. To that end, a dual reporter strain (i.e., reporters for SOS response and prophage induction) is cultivated in lab-on-a-chip devices and analyzed using fluorescence microscopy. From the time-resolved reporter outputs, we have established a cellular state model that is used for comprehensive population modeling.

Contents

Acronyms

BHI brain-heart infusion

BRIEF Binary Robust Independent Elementary Features

CFD computational fluid dynamics

FC flow cytometry

GHT Generalized Hough Transform

JuNGLE Jülich Next Generation Lineage Extractor

LAP linear assignment problem

LOC lab-on-a-chip

LP linear program

MGC monolayer growth chamber

MSCC microfluidic single-cell cultivation

PCA protocatechuic acid

PDMS polydimethylsiloxane

ROI region of interest

SCA single-cell analysis

SIFT Scale-Invariant Feature Transform

SIM Shape Index Map

SNR signal-to-noise ratio

SPI spontaneous prophage induction

SPT single-particle tracking

SURF Speeded-Up Robust Features

SVM Support Vector Machine

TED Tree Edit Distance

Symbols

$r(t)$ A closed curve in the plane, where $t \in \mathbb{R}$ is a continuous, artificial parameter

$\mathcal{R}$ The region that is enclosed by a closed contour $\bar{\mathcal{R}}$

$\bar{\mathcal{R}}$ A contour (closed curve)

$\mathcal{H}$ Hessian matrix

$\mathbf{I}$ Identity matrix

I General 8-bit image. $I(\mathbf{x}) : \Omega \to [0, 2^8)$

Ω Image domain

ε^2 Optimization criterion for pyramid approach to registration

$\mathbf{P}$ A parameter vector of a transformation Q

R A reference image that is used as the registration target

T An target image used for registration of stacks

Q A transformation of an image

$\mathbf{W}(\mathbf{x}; \mathbf{p})$ Warp of coordinates $\mathbf{x}$ with parameter vector $\mathbf{p}$

φ Basis function

M Number of control points of a parametric snake

E_{snake} Snake energy

E_{Edge} Edge energy term of a snake energy E_{snake}

E_{Region} Region energy term of a snake energy E_{snake}

α Trade-off parameter

E_S Splitting energy for two points

μ Mean of a probability density function

σ Standard deviation of a probability density function

σ^2 Variance of a probability density function

$C(x, y)$ Contrast of a window centered at (x, y): $C(x, y) = \max(x, y) - \min(x, y)$

θ A (gray value) threshold.

t_{div} Time between two cell divisions, i.e., the duration of one cell cycle

$t_{div\_detected}$ Time between two cell divisions as observed using time-lapse microscopy

Δt Time between to frames of an image sequence

A Assignment matrix for frame-to-frame linking

g Cost for closing a gap between two track segments

l Linking cost for a one-to-one assignment between two cells

C_{global} Cost matrix for global track segment linking, shown in Figure 5.4

C_{local} Cost matrix for frame-to-frame linking, shown in Figure 5.2

m Cost for merging two track segments

s Cost for creating a split event

Part I

Introduction

1 Introduction

In recent years, the general focus of many countries has shifted from a petrochemical-based to a bio-based economy. These sustainability concepts have resulted in the engineering of bacteria, yeast, and fungi to so-called (microbial) cell factories [3]. It is nowadays possible to produce a multitude of compounds, i.e., pharmaceuticals [4, 5], food additives, e.g., lysine [6], as well as biofuels [7, 8], in large-scale biotechnological processes using microbes. An important future step in the direction of sustainability, should be the coupling of biological and chemical methods for producing chemical building blocks and intermediates based on a "green" bioconversion instead of petrochemical processes.

According to literature, the global industrial biotechnology sector is expected to grow to around 515 billion euros in 2020 and further [9]. This shift to a bio-based economy is further emphasized by various global initiatives investing millions of euros (e.g., Innovation Initiative Industrial Biotechnology [10], Industrial Biotechnology Catalyst by BBSRC [11]) in industrial biotechnology and bioeconomy-related fields [12].

Driven by advances in biotechnology [13], the importance of heterogeneity of populations and cellular characteristics for processes like metabolism, growth, and production, is currently under investigation [14, 15]. In contrast to bulk experiments that are investigating processes on a population average level, specialized experiments with single-cell resolution have to be performed to yield information about cellular heterogeneity and the governing mechanisms [2, 16]. This knowledge can in turn be used for further optimization of bio-based production processes. Hence, single-cell studies are currently considered to be an additional layer of bioprocess optimization [15] (cf. Figure 1.1).

Many current studies are based on the understanding that clonal bacterial populations can be physiologically heterogeneous [17]. Single-cell investigations of microbial (production) strains have lead to unexpected discoveries, such as the existence of biological noise and phenotypic switching in persistence [18], toxicity [19], and bacterial growth [20–23]. Investigations of such phenomena have greatly profited from the progress in handling individual cells under constant conditions as well as microscopy. With

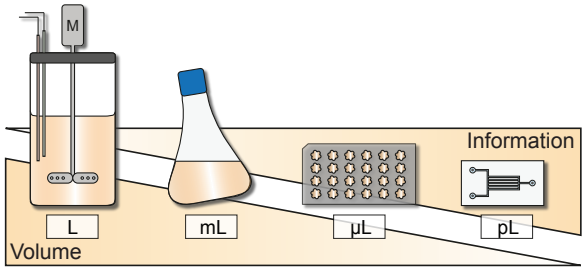


Figure 1.1: Different scales of microbial analysis. **A.** Lab-scale bioreactor. **B.** Shaking flask. **C.** Microtiter plates. **D.** Microfluidic LOC devices (adapted from [15]).

micro-scale experiments executed on LOC devices, researchers are able to run up to thousands of cultivations in parallel [2]. In combination with state-of-the-art live-cell imaging setups [24, 25] even more detailed information about microbial systems is obtained. Due to parallelization of experimental methods, large amounts of data are generated within hours. This is especially important for generating statistically sound data about rare events that happen with low frequency.

While optical measurements are helpful to determine cell numbers in a population or investigate cell morphology, fluorescent biosensors increase the dimensionality of data that is acquired for individual cells [26]. The general concept of biosensors is to construct bacterial strains with genes coding for fluorescent proteins integrated into their genome or on plasmids [27]. Those reporter genes are fused to promoters of interest such that they underly the same control mechanism as the gene of interest. In a live-cell imaging experiment, such cells will express fluorescent proteins that are excitable. The emission intensity after excitation is measured, resulting in an optical readout that is used to characterize the expression pattern of a studied gene [27].

Because the use of fluorescent reporters helps to resolve translational heterogeneity [28], it can ultimately lead to a deeper understanding of microbial processes on the systems level. Nevertheless, bulk analyses are useful for extracting trends with averaging measurements for entire microbial populations. However, to resolve the distributions that generate this average, single-cell technologies have to be applied (cf. Figure 1.2). An often referred to example is the analysis of *lac* operon induction levels in *Escherichia coli* [29]. The smooth increase in protein expression that is observed on population level is not due to a gradual induction of all cells but rather due to the formation of subpopulations that are fully induced, while the rest of the population is fully repressed.

Unlike other single-cell analysis techniques, e.g., flow cytometry, live-cell imaging approaches are able to temporally resolve data. While this opens up great opportunities

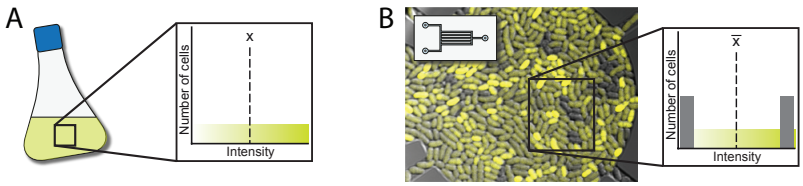


Figure 1.2: Population heterogeneity at different resolutions. **A**. Bulk measurements extract one averaged fluorescence intensity value for all cells. **B**. Single-cell analysis enables the extraction of spatially-resolved information from individual cells. The resulting population distribution has the same mean value as extracted from bulk analysis, but resolves two subpopulations.

for understanding biological processes, it also burdens the data analysis with the additional temporal dimension to be considered. With the microfluidic LOC devices made in Jülich, image sequences of up to hundreds of microfluidic single-cell cultivation (MSCC) are acquired in parallel. Hence, the LOC platform, which has been developed in the *Microscale Bioengineering* group of Dietrich Kohlheyer, creates opportunities for high-throughput, single-cell investigations. However, to be able to explore these opportunities, data evaluation workflows have to be established. The challenge remains to extract quantitative, time-resolved data from large image sequences in a robust, accurate, and automated way.

1.1 Project Objectives

When this PhD project started in 2012, the production procedure for microfluidic LOC devices made in Jülich had been established and the live-cell imaging setup with bright-field and fluorescence microscopy was already available. While the first acquired images were only analyzed qualitatively, it was clear that the quantification of image contents was required. Hence, a workflow for the extraction of quantitative cell properties (*descriptive data*) from image sequences (*raw data*) had to be established at the institute as presented in Figure 1.3.

The extraction of quantitative cell descriptors was achieved using manual selection of cells in the vendor software before this PhD project. If the extraction of time-resolved information for individual cells was desired, the assignments between subsequent images and the final visual representation had to be done manually. As already mentioned, both task are tedious, time consuming (in the order of days per cultivation chamber), and error prone.

Therefore, the objective was set to establish a (semi-)automated workflow for the extraction of time-resolved information from image data. This (semi-)automated

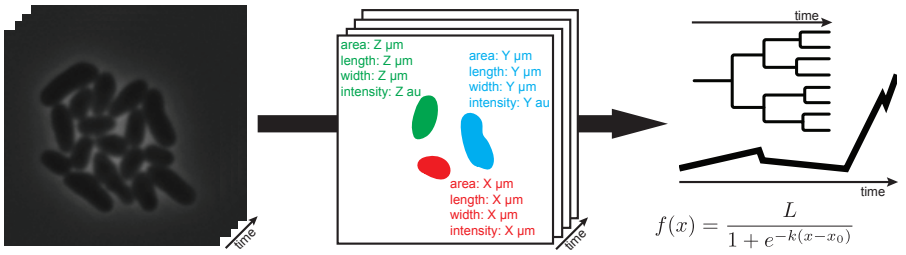


Figure 1.3: Extraction of time-resolved, single-cell information from image sequences enables the investigation of dynamics of cellular processes. Establishing the process for the automatic extraction as well as the statistical analysis of extracted data is the objective of this PhD project.

workflow should be implemented in close collaboration with experimentalists to *a*) have direct contact with the users and *b*) optimize the quality of image data for processing and analysis. It is, therefore, not an objective of this project to create a completely automated solution for information extraction that is not practically applicable to the data that is generated in our labs.

The workflow is implemented as a ready-to-use software package that helps experimentalists with analyzing acquired image data. Hence, the software has to be flexible and extensible for the various use cases encountered at the institute. A modular software architecture (with sensible interfaces) supports these requirements and opens up the software for future improvements and optimization of individual modules.

In addition to the design and implementation of the analysis workflow, a server solution for storing acquired image data in a consistent and collaboration-enabling way is desired. This solution should feature a centralized storage of images in combination with additional (experimental) information. Additionally, the software solution should be ready to act as a central processing hub. The ideal case for an experimentalist would be to directly execute analysis tasks on a server instead of their desktop computers. Such tasks could be distributed on available computation infrastructure without users noticing.

All in all, the objective on a technical level is to establish image analysis as an integral part of the microfluidics LOC platform made in Jülich and, in a broader context, the single-cell analysis (SCA) at IBG-1.

The aforementioned workflow will be applied in this work to generate additional knowledge on several topics of IBG-1. In close collaboration with the *Population Heterogeneity* group of Julia Frunzke, the connection between SOS response in *Corynebacterium glutamicum* and the prophage CGP3 is investigated. To that end, experimental data is acquired, analyzed, and used for modeling purposes in the context of the federally

funded (DFG) priority program SPP1617 “Phenotypic heterogeneity and sociobiology of bacterial populations”.

Furthermore, microbial growth in the context of process optimization and screening studies is one focus in close collaboration with Alexander Grünberger in the Kohlheyer lab.

1.2 Outline

Chapter 2 presents experimental procedures as well as additional information on LOCs devices and live-cell imaging for the generation of time-lapse image sequences. This chapter also elaborates on improvements to the data management life cycle at the IBG-1.

The implemented data extraction and image analysis workflow is described in detail in Part II. Each chapter is concerned with the theoretical background of one module of the pipeline. This part is concluded by a chapter on the implementation of the presented pipeline as an ImageJ/Fiji plugin.

Part III shows a subset of the experiments that have been evaluated using the image analysis workflow as presented in this thesis. The first chapter presents the contributions of the presented pipeline to characterizing the microfluidic chips. The subsequent Chapter 10 highlights the opportunities as well as the pitfalls of using image analysis techniques for microbial growth quantification. The following chapter describes a screening study that presents data on the connection between nutrients and the morphology of bacteria. Furthermore, a focus is on the modeling of the SOS response in *C. glutamicum* and prophage induction from experimental data in Chapter 12 using fluorescence reporters.

The final part concludes the thesis with a short summary and final remarks about the extracted insights from this project. Also, future directions and projects for the image analysis at IBG-1 are presented in Chapter 14.

2 Data Generation

Since the major objective of my PhD project is the establishment of an image analysis pipeline in the context of microfluidic SCA for *C. glutamicum*, this chapter present the already established workflows and the (experimental) infrastructure at IBG-1. While the focus in on the internal experimental workflow, most of the concepts can be transferred to other setups or are universally applicable.

Figure 2.1 sketches an exemplary investigation workflow for SCA at IBG-1. Several persons and groups are involved in the presented workflow: single-cell experiments are executed in the Kohlheyer lab to investigate a bio(techno)logical question posed by members of bio(techno)logical groups, e.g., the *Population Heterogeneity* group of Julia Frunzke. Data extraction, analysis, and modeling tasks are executed by members of the Modeling and Simulation group.

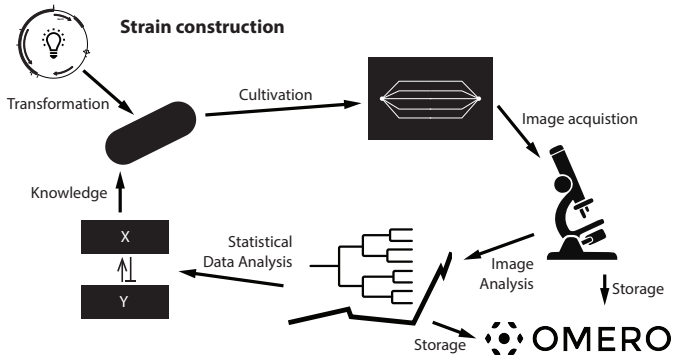


Figure 2.1: General SCA workflow. Bacterial strains that contain biosensors are engineered. Strains are cultivated in microfluidic LOC devices that are imaged in a state-of-the-art live-cell imaging setup. Image analysis techniques are used for the extraction of descriptive data, that is stored in OMERO together with raw data. Statistical data analysis and modeling generate new knowledge systems level knowledge about intracellular processes.

This chapter provides fundamental knowledge as well as background information on the individual steps that are sketched in Figure 2.1 apart from "Image Analysis" (cf. Part II) and "Statistical Data Analysis / Modeling" (cf. Part III). Starting with the platform organism that is at the center of presented applications, this chapter includes a short introduction of *C. glutamicum* that also highlights its importance as a versatile microbial cell factory and why it is subject of basic research. An overview of the microfluidic LOC platform as well as the live-cell imaging setup is provided to show the origins of the raw data that is to be processed with the image analysis pipeline. Finally, this chapter describes the storage solution for acquired image sequences that has been established in this project.

2.1 *Corynebacterium glutamicum*

Corynebacterium glutamicum is a gram positive, non-pathogenic soil bacterium that is one of the major platform organism for the biotechnological production of various amino acids, e.g., L-glutamate, L-valine, and other food and feed additives, e.g., nucleic acids and vitamins, at industrial scale. For example, *C. glutamicum* is used in a highly optimized biobased process to produce L-valine, a food additive, in the order of million tons per year [30]. In the last decade, *C. glutamicum* has also been engineered for new applications in the "white" biotechnology. Strains have been developed and optimized for the production of biofuels [8] as well as organic acids [31, 32]. Although briefly mentioned in the introduction, the contribution of *C. glutamicum* in health-related fields is mostly due to the usage of amino acids as pharmaceutical raw material. Additionally, *C. glutamicum* is subject to research as it is a close relative to the pathogenic *Mycobacterium tuberculosis* [33].

For an overview of the production capabilities of *C. glutamicum*, the reader is referred to the comprehensive introductory chapter "The Biotechnological Potential of Corynebacterium glutamicum" of [34] as well as [30, 35].

2.1.1 Morphology, Growth, and Division Behavior

C. glutamicum belongs to the family of rod-shaped bacteria. The typical wildtype strains range from cell lengths of $2\,\mu\mathrm{m}$ to $5\,\mu\mathrm{m}$, while the width is about $1\,\mu\mathrm{m}$ [36]. The general morphology of cells is shown in an scanning electron microscopy image of a small population in Figure 2.2A.

In contrast to other rod-shaped bacteria, *Actinobacteria* including *Corynebacteria* grow by apical insertion of peptidoglycan into the cell wall [37]. Which components play an integral role in the apical growth machinery and how it is located, is a topic of active research [38, 39].

C. glutamicum shows a "snapping" division behavior that is illustrated in Figure 2.2B. Cells will abruptly (in less than $1\,\mu\mathrm{s}$) move post-fission, effectively creating a V-shape

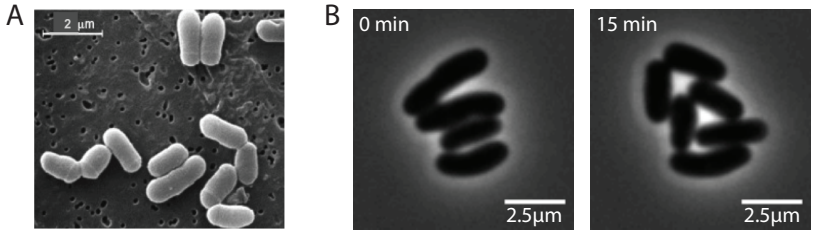


Figure 2.2: Cellular shape and division behavior of *C. glutamicum*. **A.** A scanning electron microscopy image of several *C. glutamicum* cells (taken from [30]). **B.** Excerpt from an image sequence of a small *C. glutamicum* population. Time between image acquisitions $15\,\mathrm{min}$. The first image shows four cells. The second image shows six cells with four in V-shaped position directly after cell division.

that is characteristic for *C. glutamicum*. Such division behavior has previously been described in *Mycobacteria* [40, 41]. A detailed understanding of the mechanisms that drives the snapping in *C. glutamicum* is not available yet. There are, however, advances in understanding cell morphogenesis [37] and the resulting V-snapping [42] in recent years.

In contrast to the cell division of other model organisms, e.g., *E. coli* and *Bacillus subtilis*, the V-snapping and the attachment of cells after division in *C. glutamicum* poses challenges for major parts of the image analysis pipeline. Detection of individual cells is hard to achieve because of the stickiness of cells post-fission making the separation of microbial populations a challenging task. What is even more challenging is finding correspondences between subsequent images, when the majority of cells in a population are dividing in a snapping manner at the same time. In this case, depending on the location in the population, cells might be moved farther than two-fold of their cell length between two subsequent frames.

2.1.2 Bacteriophages

The *C. glutamicum* chromosome has a total number of approximately 3.3 Mbp (cf. Figure 2.3). Analyses of the sequenced chromosome have identified regions with putative bacteriophage origin [43, 44]. The CGP1, CGP2, CGP3, and CGP4 termed prophages (i.e, a bacteriophage genome integrated into the host chromosome) differ significantly in their contribution to the *C. glutamicum* genome. In contrast to the smaller CGP1, CGP2, and CGP4, the CGP3 element is one of the largest known prophages (∼187 kbp), constituting ∼6% of the entire *C. glutamicum* genome.

Such prophages are bacteriophage genomes that have been integrated into the host genome [45]. Hence, the *C. glutamicum* strain, that we are investigating has already been infected by various bacteriophages. Upon injection into a host cell, the phage

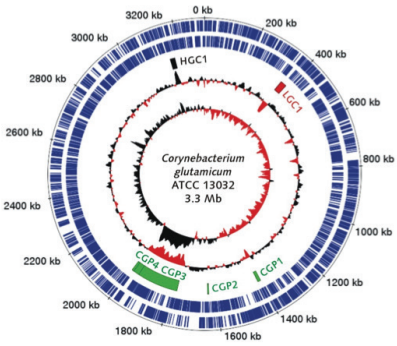


Figure 2.3: Genotypic characteristics of *Corynebacterium glutamicum*. Genome sequence published in 2003 [43]. The 3.3 Mbp genome contains four regions of putative prophages denoted by CGP1-4, which together make up ∼10% of the genome.

integrates into the host genome and stays dormant for longer periods of time (cf. "temperate prophage" in Figure 2.4). During this lysogenic life cycle, stresses trigger the switching from the lysogenic to the lytic lifecycle [45–47]. In the lytic life cycle, the bacteriophage replicates in the host cell and is released when the host is lysed.

In preliminary flow cytometry (FC) experiments, we have observed that the prophage CGP3 is induced in absence of external stress (e.g., antibiotics) in a minority of *C. glutamicum* cells. While CGP3 can not be packaged into a fully functional phage, the induction apparatus seems to be stochastically triggered. We will focus on those findings in Chapter 12.

2.1.3 Fluorescence Reporters

Several fluorescence reporters for various applications are commercially available. Since *C. glutamicum* is a microbiological model organism, standard cloning procedures to integrate fluorescence reporters into cells are well established [30]. There are several possibilities for the integration of reporters into a host. One possibility is the direct integration into the genome of the host organism as fusion protein (i.e., reporter gene and gene of interest are joined) or as "standalone" gene with the promoter of interest. Second is the generation and introduction of a plasmid (cf. Figure 2.5) that contains the genes for the fluorescent reporters under control of the target promoters [26].

Fluorescence reporters differ in their excitation and emission wave length and are dubbed according to the part of the spectrum where their emission wave length is located. Yellow fluorescent proteins have their emission peak at around $530\,\mathrm{nm}$, while red fluorescent proteins are located in the higher energy region of the wave length

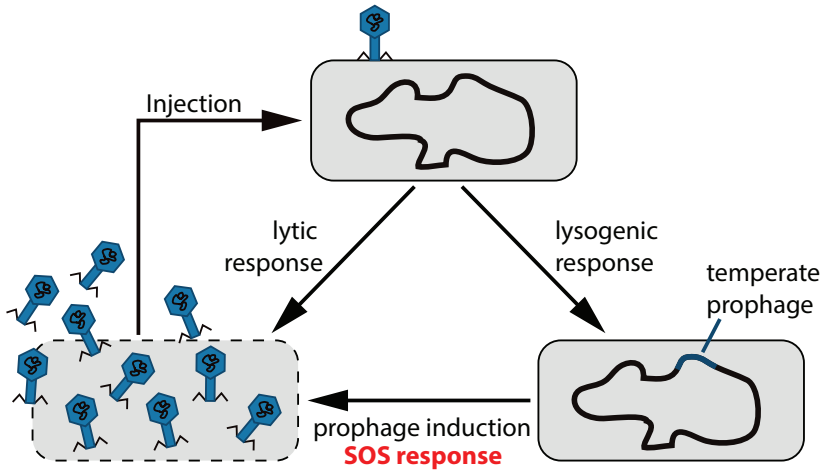


Figure 2.4: Schematic of the life cycle of bacteriophages (adapted from [45]). Upon injection into a bacterial host, bacteriophages can enter the lytic or the lysogenic life cycle. The lytic lifecycle will kill the host to produce more bacteriophages for continuation of infection. Bacteriophages can also enter a dormant state in which they are integrated into the host genome. A switch between the states is possible.

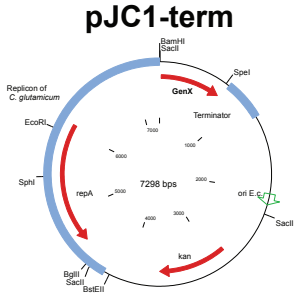


Figure 2.5: The plasmid pJC1-term that has been used for the transformation of *C. glutamicum* in the studies of presented in this thesis. **Gen X** denotes the gene that is introduced into the organism.

spectrum (over $600\,\mathrm{nm}$). Additionally, many improvements have been achieved in the field of fluorescent protein design to improve the signal strength as well as other properties.

The reporters used in the applications presented in Part III are Venus (red, [48]) and E2-Crimson (yellow, [49]). E2-Crimson and Crimson will be used synonymously

throughout the thesis unless stated otherwise. Both fluorescent proteins have been selected such that the excitation as well as the emission wave length are well separable during fluorescence microscopy (cf. Figure 2.6).

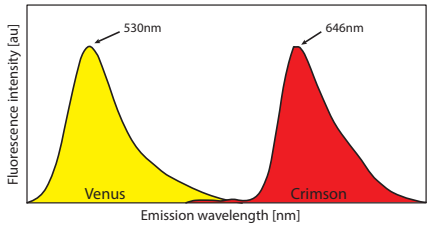


Figure 2.6: Schematic of emission wavelengths of Venus and E2-Crimson.

In live-cell imaging, fluorescent proteins are most commonly employed to track dynamics of proteins (if temporal resolution allows). To that end, a sequence coding for a fluorescent protein is placed under transcriptional regulation of the promoter belonging to a gene of interest. Hence, this approach provides a directly visible readout of the expression pattern of the gene of interest in the living organism. This is in stark contrast to dyes where foreign molecules need to be introduced to the host.

2.2 Microfluidic Single-Cell Cultivation

Originally LOC devices have been developed to miniaturize standard laboratory processes to a scale where several processes are executed on a chip of the size of a coin. Figure 2.7 shows such a LOC device manufactured from polydimethylsiloxane (PDMS). This chip consists of channels of tens of micrometers diameter, and even below, through which a fluid is pumped with constant flow rate. Such microfluidic LOCs offer the potential for different applications in all fields of the life-sciences [15, 50–54]. Drug development, bioanalyses, and cultivation of single-cell samples are only a few possible applications within microfluidic systems.

Such microfluidic LOC devices are especially important for the investigation of microbial population heterogeneity. When bacteria are cultivated in shaking flasks at lab-scale, cells are subject to environmental changes due to shaking or stirring. This effect is especially significant in batch processes, where the nutrient supply changes over time and side products might accumulate. These permanently changing environments contribute to cellular heterogeneity in a cultivation. If the objective of an experiment, however, is to study heterogeneity due to inherent stochasticity of cellular processes, environmental influences have to be eliminated as far as possible. This is hardly possible for large-scale experiments due to mixing constraints. This also holds for cultivations on agarose pads for single-cell investigations [55], where the environment of individual cells will slightly change over time. Mostly, due to nutrient uptake but also due to

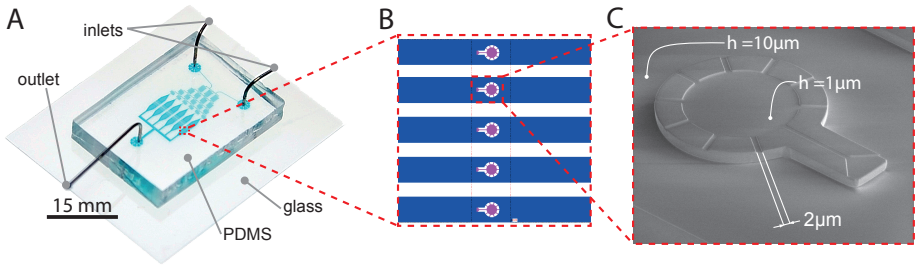


Figure 2.7: The fabricated microfluidic chip device at different levels. **A.** A PDMS chip bonded to a 170 µm glass slide. **B.** An excerpt from the blueprint of the chip. Five parallel channels with a bioreactor each. **C.** Scanning electron microscopy image of one cultivation chamber (adapted from [2]).

the excretion of (side)products into the environment. Besides nutrients, the cellular environment comprises additional parameters, like temperature, pH value, and oxygen supply, that directly impact intracellular processes.

Microfluidic LOC devices for MSCC work according to the principle of constantly flushing fresh medium through the devices. The flushing of the chip helps to keep the nutrient situation in growth chambers constant over time. The general idea behind single-cell cultivation is to catch individual cells in specifically engineered and manufactured growth sites. In combination with an incubator the temperature is precisely controlled. The incubator, furthermore, allows for the control of oxygen concentration as well as nitrogen concentration during an experiment. Hence, growth sites are state-of-the-art structures in which clonal population can develop.

The overall chip design as well as the design of the growth sites is determined by the respective application. The growth sites that have been used for the studies in this thesis can harbor microbial populations of several generations (up to 10). The microfluidic devices used for this project are intended for the growth analysis of bacterial cells on single-cell level. In contrast to agarose pads, data are obtained in high-throughput in microfluidic devices [56]. To that end, the number of parallel cultivations is raised (cf. Figure 2.7B) as well as the degree of automation. A typical LOC device contains hundreds of growth sites. Figure 2.8 shows two cultivation chambers that are used to grow clonal populations from individual cells. While many designs have been proposed for MSCC [57, 58], the focus in this thesis is on the rectangular monolayer growth chamber (MGC) shown in Figure 2.8. At some time during the experiment, the size of this population will exceed the space that is available in the chamber and cells will be pushed out of the chamber through the so-called overflow channels.

Although the whole design of the described cultivation chambers is complex, the role of the height of such a chamber should be highlighted. In order to obtain quantitative,

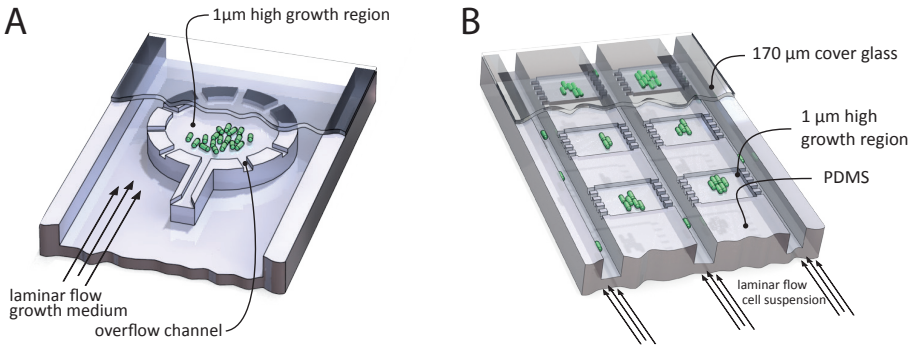


Figure 2.8: Growth sites for cultivation of bacteria. Cells are growing to a full population inside the chamber. Once the chamber is filled with cells, the surplus exits through the overflow channels into the supply channel. This deeper channel is constantly flushed with medium and provides the population with nutrition. **A.** The pico-litre bioreactor (illustration adapted from [2]). **B.** Several MGCs with the respective main channels.

time-resolved information for each cell, cells have to be visible over the course of an experiment. While the height of MGCs varies between $800\,\mathrm{nm}$ and $1.2\,\mu\mathrm{m}$, it is guaranteed that cells do not overlap to their full extent but only partially, if at all. Since the effect of overlapping can not be eliminated entirely in the presented bioreactor, this challenge has to be approached with image analysis means (cf. Section 4.6). Details of the fabrication process and the mode of operation are presented in [2] and [59].

Complementing the MGC and other population growth chambers, designs like the mother-machine [60] have been proposed to cultivate individual cells for extended periods of time. The possible cultivation time in the mother-machine is increased at the expense of population information. The mother-machine is used as an alternative to population-based growth sites at IBG-1, when population information can be discarded or population influences are unwanted. Since this is not the case for the applications presented in this work, focus of this thesis is on MGCs.

2.2.1 Operating Microfluidic Devices

Although the standard operating protocols for MSCCs differ in details, a common scheme can be defined on how microfluidic LOCs are operated in a live-cell imaging experiment.

Setup Syringes are filled with medium. The syringes are connected to a syringe pump as well as the chip. The pump will push the medium through the chip at constant speed.

Seeding To seed bacteria into the traps/growth sites, medium that is charged with precultivated bacteria is flushed through the chip. In order to increase the seeding efficiency, a special procedure [61], which has been optimized with image analysis techniques, is applied.

Cultivation After the initial seeding phase, fresh medium (without bacteria) is flushed through the chip for the complete runtime of an experiment.

2.3 Live-Cell Imaging Setup

A fully motorized, inverted epifluorescence microscope (Ti-Eclipse, Nikon GmbH, Düsseldorf, Germany) is used to obtain high resolution time-lapse images (shown in Figure 2.9). Due to the motorized stage, different predefined locations on a chip are imaged in a completely automated manner (cf. Figure 2.10). Fundamental bright field, fluorescence, and phase-contrast microscopy are possible with the technical equipment.

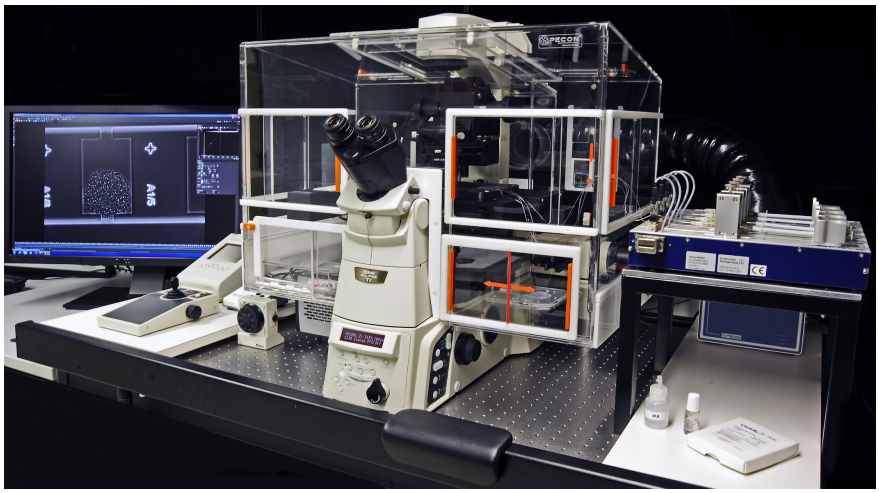


Figure 2.9: Live-cell imaging setup. In the center a Nikon Eclipse Ti with a stage surrounded by an incubator keeping environmental conditions (temperature, oxygen supply, etc.) constant. A system of syringe pumps that are connected to the LOC is located on the right side. Controls for the microscope are found to the left of the microscope right next to the computer used for setting up experiments.

Phase-contrast microscopy is one of the most widely used imaging techniques in live-cell imaging. It is based on the conversion of phase shifts of light passing through a specimen, to brightness changes in the resulting image. Especially when working with unstained biological specimen, the contrast between a cell and the surrounding

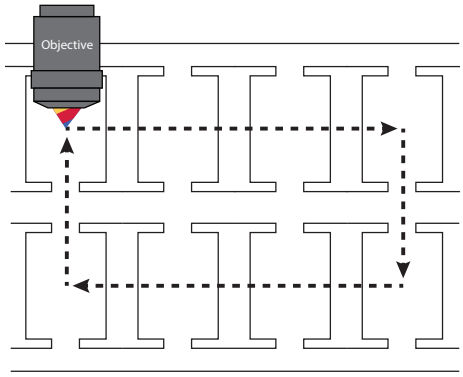


Figure 2.10: Acquisition scheme of images with the experimental setup.

is low when using bright field illumination. Comparable techniques like differential interference contrast (DIC) microscopy are also used for live-cell imaging, but show weaknesses, like reduced contrast and visibility of cells [62].

An oil-immersion objective (Apo $100\times$ Oil N) with a numerical aperture (NA) of 1.45 and a magnification of 100x is used for the experiments. According to Abbe's law, the resolution limit r of the used system is at approximately $200\,\mathrm{nm}$:

$$r = \frac{\lambda}{2 \times NA}, \tag{2.1}$$

with wavelength λ and the numerical aperture of the objective NA.

In addition to phase-contrast images with an LED light source (pE-100 white, CoolLed Ltd., Andover, UK), the microscope can also generate fluorescence images using a Nikon fluorescence excitation light source (Intensilight). Images are acquired in combination with high quality filters for Venus (EX 520/30 nm, DM 510 nm, EM 540/20 nm) and E2-Crimson (EX 600/37 nm, DM 630 nm, EM 675/67 nm; AHF Analysentechnik AG, Tübingen, Germany).

The used CCD cameras (Clara DR-3041 and Neo sCMOS, Andor Technology Plc., Belfast, United Kingdom) generate images with a spatial resolution of $1600\,\mathrm{px} \times 1200\,\mathrm{px}$ and $2560\,\mathrm{px} \times 2160\,\mathrm{px}$, respectively. Hence, images of common growth sites are about $800\,\mathrm{px} \times 800\,\mathrm{px}$ in size. At the aforementioned resolution, an image sequence of a single cultivation chamber at 16-bit, consisting of 120 images, takes up $450\,\mathrm{MB}$ to $1300\,\mathrm{MB}$ of disk space. Since one lab-on-a-chip device features several hundreds of cultivation chambers, the data generated per month is in the order of terabytes.

While it is theoretically possible with the technical setup to resolve differences in objects down to the resolution limit of $200\,\mathrm{nm}$, the observation of bacteria with lengths between $2\,\mu\mathrm{m}$ and $4\,\mu\mathrm{m}$ remains challenging. A higher resolution of the acquired images leads to more refined cell contours because they are sampled more of often and, hence, consists of more pixels. Benchmarks of the influence of imaging resolution on cell detection results are provided in Chapter 7.

2.4 Central Storage for Biological Image Data: OMERO

The integrated NIS-Elements software (Nikon GmbH, Düsseldorf, Germany) is used for controlling the image acquisition setup. Experimental output is stored in the proprietary file format ND2 [63]. Such files contain, in addition to the raw image data additional information, for instance, experimental settings, like position of an acquired image, and settings like exposure time. This additional information is termed meta information. With the NIS-Elements software, a comfortable image viewer is available, that can load even large ND2 files from an external location, like a network share.

At IBG-1, experimentalists have started to create a folder structure of ND2 files to keep experimental data in order. To have the files sorted according to the date of experiment, the structure usually looks like:

```
USERNAME
+-- YEAR-MONTH-DAY_EXPERIMENT_NAME
+----- EXPERIMENT1.nd2
+----- EXPERIMENT2.nd2
+----- EXPERIMENT3.nd2
```

While this folder structure serves the purpose of organizing data of an individual reasonably well, it is not feasible for collaborative experiments and analyses. In case of collaborations, multiple people are part of the data generation and analysis workflow Figure 2.1. Common practice in this case is the duplication of the experimental data, which leads to inconsistencies in the worst case.

In addition to the organization of image data, experimental meta data are lost, when only ND2 files stored. This information is namely *a*) the cultivated organism, *b*) cultivation medium, *c*) protocol of preparation procedure, and *d*) notes and remarks to the experiment. This is crucial information about imaging data, which is intrinsically tied to the images, and has to be stored as annotations, for instance.

The aforementioned tasks are of general nature and have been tackled with great efforts by the recently established Open Microscopy Environment (OME) consortium. The consortium works in close contact with laboratories and imaging facilities around

the world to decide on a consensus for storing biomedical image data. One result of this process is the client/server application OME Remote Objects (OMERO) for storing biological image data in connection with available meta data.

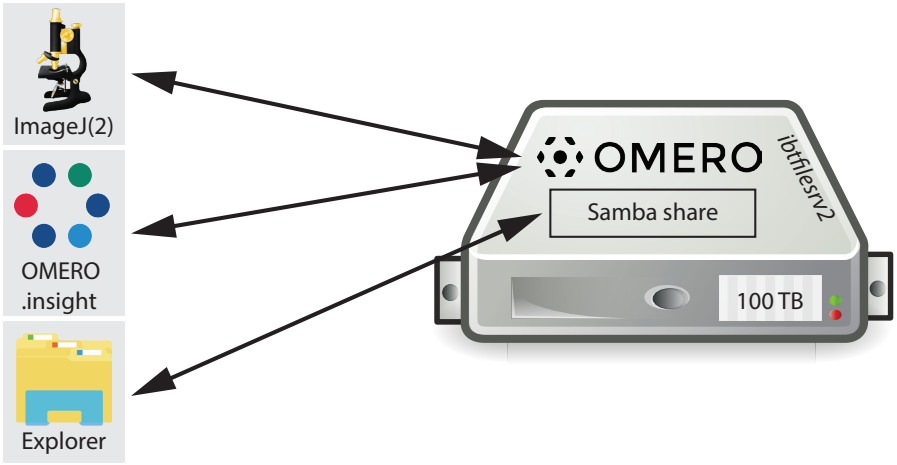


Figure 2.11: Overview of OMERO infrastructure at IBG-1. Large boxes denote dedicated machines, clients as well as servers. Arrows denote interactions between clients and the server. Clients architecture is heterogeneous, Windows, and Linux.

Figure 2.11 shows the heterogeneous infrastructure that has been established at the IBG-1 during the time of my PhD project. In addition to storing image data combined with meta and experimental information, a long term goal is to foster collaborations using the data sharing features. Furthermore, the computer *ibtfilesrv2* is designed to be the central machine for a network of distributed compute nodes that are used for demanding image processing and analysis tasks.

Part II

Image Analysis

Image Analysis of *C. glutamicum*

The extraction of quantitative or qualitative, time-resolved information from time-lapse image sequences requires advanced image processing and analysis approaches. On the one hand, images need to be processed so that they can be more easily interpreted. On the other hand, an image analysis pipeline for the extraction of information has to be established.

The reader might ask the question of the necessity of yet another image analysis pipeline. This question is easily answered in a broader context. State-of-the-art image processing tasks require highly optimized approaches for very specific problems [64]. Following this paradigm, general image processing solutions are barely available or under perform with real data with respect to accuracy, computational time, or computational demand. While these issues might be of technical origin, the additional challenges in biological image analysis are peculiarities of the imaged biological specimen.

It is these peculiarities of *C. glutamicum* in combination with the microfluidic setup that requires adaptations of available algorithms to fit the needs of experimentalists. Three of those peculiarities will be highlighted in the following chapters to argue for the necessity of refining available approaches:

1. microfluidic device structures that are not of interest,
2. dense microbial populations in MGCs, and
3. unpredictable division behavior of *C. glutamicum*.

The processing workflow established during this PhD project is separated into several subsequent image processing and analysis steps (cf. Figure 2.12). In contrast to recent developments [65], a sequential approach that contains several submodules was implemented. These submodules will be described in detail in the following chapter. This procedure allows for re-using modules and optimization of the pipeline for different

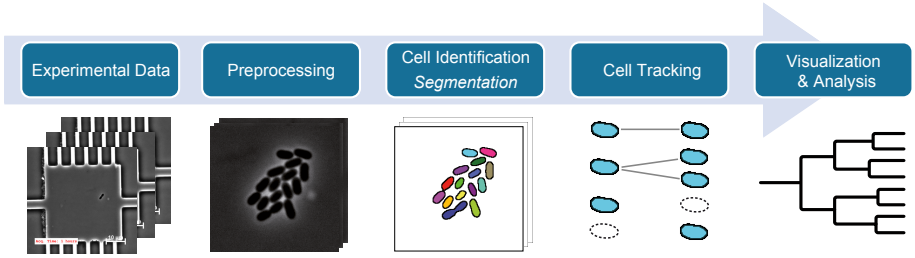


Figure 2.12: Schematic of an image analysis pipeline for the extraction of time-resolved data. Raw images (generated as described in Chapter 2) have to be preprocessed to remove unnecessary chip structures and spatial shifts in image sequences. To extract quantitative information, cells are detected in each image of a sequence. Correspondences are subsequently established. Finally, the time-resolved information is visualized and structured in lineage trees that are also used for further postprocessing.

applications. This would not be possible with an integrated approach comprising all steps of a pipeline.

In the first step of the pipeline, raw input image sequences have to be prepared for further processing. Growth sites are determined as regions of interest (ROIs) and are cropped to discard structures that are not of interest and speed up subsequent computations. The **preprocessing** also removes spatial shifts between subsequent images (also called *frames*) in an image sequence. Such shifts occur regularly due to the applied image acquisition scheme (cf. Figure 2.10).

The concept of **image segmentation** is to transform an image such that it is easier to analyze and quantify with respect to specific research questions [66]. Hence, the general objective is to assign each pixel in an image a label such that pixels that share a specific characteristic are assigned the same label. We are, however, not only looking for the best segmentation of an image but for context awareness. That is, the objective is to identify individual cells and their contours in a population rather than only connected regions.

After cell detection, it is possible to extract quantitative information for individual cells in a clonal population. However, it is not possible to investigate dynamics without time-resolved data. Depending on the imaging frequency in the time-lapse microscopy setup, an individual cell is imaged multiple times during a cell cycle. Finding the correspondence of each cell throughout a sequence of images is called **tracking** [67].

The quantitative information that has been extracted for cells, finally, needs to be combined with the development information that has been extracted in the cell tracking step. One possible solution is the visualization of cellular properties over time (the

so-called *traces*) in line plots. This requires an information visualization approach for the data as well as an elaborate data model that can be accessed for advanced analysis tasks (e.g., machine learning techniques, information extraction, parameter extraction for simulation studies).

The following chapters will highlight approaches to solve the aforementioned tasks and challenges of an image analysis pipeline for the extraction of time-resolved, quantitative data from time-lapse microscopy. The chapter is concluded by details of the implementation of the presented modules and the image analysis pipeline as a whole.

3 Preprocessing

While image sequences that have been acquired with state-of-the-art live-cell imaging setups are of high quality, raw images cannot be used directly as input to cell detection procedures. In the particular case of microfluidic LOC devices, two challenges have to be tackled originating from the experimental setup. First, a spatial shift in xy-plane that is due to the image acquisition scheme. Second, the detection and cropping of individual growth sites for removing unnecessary image information for the subsequent processing steps.

This chapter presents approaches that have been implemented as modules of the image analysis pipeline for the conversion of raw images to input images for subsequent steps.

3.1 Registration

Because of the microscopic and experimental setup, in which not only one, but multiple positions are observed in parallel, the stage of the microscope moves between two subsequent acquisitions of one particular position (cf. Figure 2.10). Although this movement is rather small ($1\,\mu\mathrm{m}$ to $3\,\mu\mathrm{m}$) in comparison with the overall size of the image, time-resolved analyses require compensating for those spatial shifts (see Figure 3.1 for an example). While this shift will influence the results of the cell assignment step (that takes into account the spatial location of cells), cell detection is independent of the position of a cell. Since finding assignments between cells is a demanding task on its own, spatial shifts are not desirable.

On an abstract level, approaches to solve the image registration or image alignment problem can be divided into two categories. Feature-based methods overlay one image (*target*) onto another (*reference*) such that prominent features from both image are aligned. In order to use such methods, characteristic features have to be determined in the reference as well as the target image. Commonly, geometric structures like corners and edges are used, however, the extraction and computation of more sophisticated features is still a field of ongoing research (e.g., Scale-Invariant Feature Transform

(SIFT) [68], Speeded-Up Robust Features (SURF) [69], Binary Robust Independent Elementary Features (BRIEF) [70]).

In contrast, intensity-based methods are maximizing the similarity between two images by computing correlation metrics. Hence, major parts of subsequent images have to be (gray value) constant. If this is not the case, such approaches get fragile and solutions for the posed optimization problem might be hard to find. Furthermore, methods relying on intensity values are rather unstable for cases where the differences in illumination are striking in the images that are to be registered.

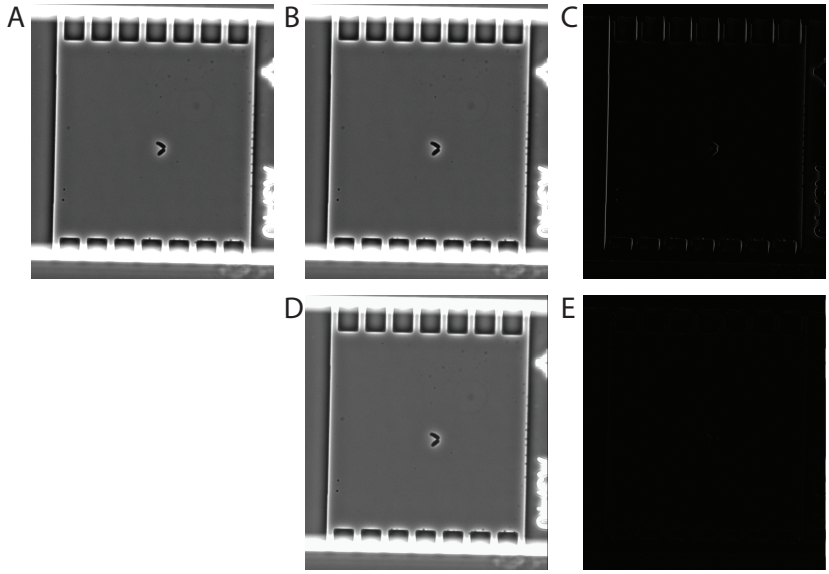


Figure 3.1: Spatial shift between two subsequent frames. **A.** Frame f_0. **B.** Frame f_1. **C.** Difference between f_0 and f_1 revealing registration errors. **D.** Frame f_1 registered to frame f_0 creating $f_{1,registered}$. **E.** Difference between f_0 and $f_{1,registered}$. Only minor differences around the two cells in the center are visible.

Image registration algorithms share the four common steps:

1. Feature detection
2. Feature matching
3. Transformation model estimation
4. Image resampling and transformation

Although the aforementioned approaches are different in their underpinnings, they share the common necessity to find suitable transformations of the *target* to align it to a *reference*. In general, the target can be transformed in several ways, e.g., scaling, rotation, translation, or shearing. Due to the large number of possible transformations, finding the best one is computationally demanding.

However, we know that the reason for the spatial shift between two images is the movement of the microscopes stage in x,y-direction (cf. Figure 2.10). With this explanation in mind, we can safely exclude scaling as well as shearing from the set of "applicable" transformations. Hence, we are only considering translation and rotation to overlay the structures of the MGCs in two subsequent frames.

While we have evaluated several approaches for solving the registration problem of MGCs, the following sections will only present the two most promising ones: the Lucas-Kanade algorithm and the approach described by Thevenaz *et al.* [71]. The reader is referred to two excellent reviews on image registration for further information and a broader overview of the field [72, 73].

3.1.1 Lucas-Kanade Algorithm

The basic idea of the Lucas-Kanade algorithm [74] is to align a target image $T(\mathbf{x})$ with a reference image $R(\mathbf{x})$, where $\mathbf{x} = (x, y)^T \in [0, H) \times [0, W) = \Omega$ is a vector of pixel coordinates with H and W the image height and width, respectively. The operations which are allowed during the alignment procedure are noted as a set of warps $\mathbf{W}(\mathbf{x}; \mathbf{p})$, with $\mathbf{p} = (p_1, p_2, \ldots, p_n)^T$ being the vector of parameters for the operations. An element of the set $\mathbf{W}(\mathbf{x}; \mathbf{p})$ takes pixels of the target image T and maps them to sub-pixel locations in the coordinate frame of the reference R. Hence, a generic set of transformations that features rotations, scaling, and translations, is formulated as:

$$\mathbf{W}_{complex}(\mathbf{x}; \mathbf{p}) = \underbrace{\begin{pmatrix} \cos p_1 & -\sin p_1 \\ \sin p_1 & \cos p_1 \end{pmatrix}}_{\text{Rotation}} \underbrace{\cdot p_2 \cdot}_{\text{Scaling}} \begin{pmatrix} x \\ y \end{pmatrix} + \underbrace{\begin{pmatrix} p_3 \\ p_4 \end{pmatrix}}_{\text{Translation}} \tag{3.1}$$

Using the available set of warps, the goal of the Lucas-Kanade algorithm is to minimize the sum of the squared error between the reference image R and the target image T warped onto the coordinate frame of the reference. Thus, the goal can be formulated as a non-linear optimization problem:

$$\hat{\mathbf{p}} = \underset{\mathbf{p}}{\operatorname{argmin}} \sum_{\mathbf{x}} \left[R(\mathbf{W}(\mathbf{x}; \mathbf{p})) - T(\mathbf{x}) \right]^2 \tag{3.2}$$

The optimization problem is solved with a Gauss-Newton gradient descent approach. An overview of the approach is provided in Algorithm 1. A through evaluation of the

Lucas-Kanade framework, the optimization problem, and the computational demand is available in a review paper by Baker and Matthews [75]. This review also considers improvements to the original formulation and implementation of the algorithm [74].

1: $\mathbf{p} \leftarrow$ initial guess
2: $\Delta\mathbf{p} \leftarrow \infty$
3: **while** $||\Delta\mathbf{p}|| > \epsilon$ **do**
4: Warp T with $\mathbf{W}(\mathbf{x};\mathbf{p})$ to compute $T(\mathbf{W}(\mathbf{x};\mathbf{p}))$
5: Compute error image $R(\mathbf{x}) - T(\mathbf{W}(\mathbf{x};\mathbf{p}))$
6: Warp the gradient ∇T with $\mathbf{W}(\mathbf{x};\mathbf{p})$
7: Evaluate the Jacobian at $(\mathbf{x},\mathbf{p})$
8: Compute steepest descent images
9: Compute the Hessian matrix
10: Compute $\Delta\mathbf{p}$
11: $\mathbf{p} \leftarrow \mathbf{p} + \Delta\mathbf{p}$
12: **end while**

Algorithm 1: The Lucas-Kanade algorithm as presented in [74].

3.1.2 Pyramid Approach to Subpixel Registration

As already stated in [75], the original Lucas-Kanade algorithm can be improved in a variety of ways. While we have used an implementation by Kang Li [76] during the first months of this project, the required robustness was not given. The problem was the constant updating of the reference image, which propagated misalignments throughout the sequence. Hence, we have moved to an implementation that is based on the work of Thévenaz *et al.* [71].

While the general scheme of the Lucas-Kanade algorithm for finding the best fitting transformation to register two images is also applied in the presented approach, significant improvements in three parts of the scheme are presented: *a*) improvements of the optimization scheme (adapted Marquardt-Levenberg (ML) algorithm), *b*) integrations of multiresolution image pyramids into the computational scheme of ML, and *c*) improved image interpolations based on cubic splines .

The optimization criterion ε^2 is formulated as $\varepsilon^2 = \|R(\mathbf{x}) - Q_{\mathbf{P}}(T(\mathbf{x}))\|^2$ where $Q_{\mathbf{P}}(T)$ is a transformation of an image T parametrized by $\mathbf{P}$, R is the reference image, and T denotes the target image. This formulation of the optimization criterion already emphasizes the intensity-based nature of the ansatz. The proposed method takes into account all the available image information (i.e., R and T) by treating each pixel intensity value as a feature [71]:

$$\hat{\mathbf{P}} = \underset{\mathbf{P}}{\operatorname{argmin}}\ \varepsilon^2 = \underset{\mathbf{P}}{\operatorname{argmin}}\ \|R(\mathbf{x}) - Q_{\mathbf{P}}(T(\mathbf{x}))\|^2 \tag{3.3}$$

We are interested in finding an affine transformation $Q_{\mathbf{P}}(T)$ that is defined by a 3×3 (rotation) matrix $\mathbf{R}$, a translation vector $\mathbf{t}$, and a gray level scaling factor γ. While $\mathbf{R}$ can in principle also incorporate skewing or shearing, we are only interested in rotations around the center of the image. Hence, we are seeking the ε^2 optimizing transformation $Q_{\hat{\mathbf{P}}}$ with $\hat{\mathbf{P}} = (\hat{\mathbf{R}}, \hat{\mathbf{t}}, \hat{\gamma})$.

The traditional ML optimization scheme searches for $\hat{\mathbf{P}}$ such that $\frac{\partial \varepsilon^2(\mathbf{P})}{\partial \mathbf{P}} = 0$. This is done in an iterative scheme where in each step an update component is computed that is then applied to the previous step:

$$\mathbf{P}_{t+1} = \mathbf{P}_t + \delta \mathbf{P}_t \tag{3.4}$$

In order to compute the update component $\delta \mathbf{P}_t$, the gradient of ε^2 has to be computed with respect to each component of $\mathbf{P}$. A recomputation of the gradient in each step of the algorithm is required, due to its dependence on the previous step. The modified Marquardt-Levenberg (ML*) algorithm, however, will in each iteration not try to find parameters $\mathbf{P}_1$ such that $\|R - Q_{\mathbf{P}_1}(T)\|^2 < \|R - Q_{\mathbf{P}_0}(T)\|^2$ but instead will find parameters $\mathbf{P}_2$ such that $\|Q_{\mathbf{P}_0^{-1}}(R) - Q_{\mathbf{P}_2}(T)\|^2 < \|Q_{\mathbf{P}_0^{-1}}(R) - T\|^2$, where $Q_{\mathbf{P}_0^{-1}}$ is the inverse transformation of the initial guess applied to the reference image.

With this new strategy the gradient of criterion ε^2 with respect to $\mathbf{P}_2$ is independent of the previous results $\mathbf{P}_i$ and is, hence, computed around a fixed point in the parameter space. In summary, the inverse transformation $Q_{\mathbf{P}^{-1}}$ that is applied to R is updated instead of the direct transformation $\mathbf{P}$ that is applied to T. For further details on the modification, the reader is referred to the original publication [71].

Furthermore, instead of using commonly applied nearest neighbor interpolation (e.g., bilinear) methods [77, p. 87-90], a cubic cardinal spline is well suited for processing data that are approximately bandlimited by nature. The advantages of using this kind of interpolation are amplified by the fact, that each evaluation of the criterion ε^2 involves a transformation of the coordinate space of an image to the reference space which is equivalent to an interpolation operation.

In addition to that, the approach proposed by Thévenaz *et al.* incorporates a multi-resolution scheme into the optimization procedure. The multi-resolution image pyramids that are subject to transformations are constructed from cubic splines (as described in the the previous paragraph). Due to the iterative nature of the ML* (as well as the original ML) it is possible to apply a hierarchical scheme. Here, the idea is to run the ML* iterations on different levels of the image pyramid and propagate parameters between the levels to the next step. The upside of this approach is, that most iterations are executed on the coarsest level of the pyramid while only moving onto more detailed levels for more fine-grained changes. This procedure decreased the overall runtime of the procedure immensely while optimizing the convergence properties.

3.1.3 Conclusions

The spatial shift and rotation that occurs within an image sequence is based on the acquisition scheme for time-lapse imaging. The stage of the microscope moves between subsequent images (in the same channel) with a precision of about $2\,\mu\mathrm{m}$. Image registration techniques [72] are applied to counter this unwanted effect to improve the quality and robustness of following processing steps.

Now that the acquired image sequence is registered, it is possible to detect the growth region of the MGC in the previously used reference image. Because of the static nature of the growth site, it is then possible to directly transfer the extracted region of interest to subsequent images. The detection of growth sites will be discussed and a solution will be presented in the following section.

3.2 Growth Site Detection

While the available information from chip structures can be used to register time-lapse image sequences, the data is not of practical interest when detecting and tracking cells. Therefore, it is desired to crop the growth site of MGCs from the original image and continue processing with the extracted ROI. Figure 3.2 provides an overview of the procedure with the additional step of rotating the ROI.

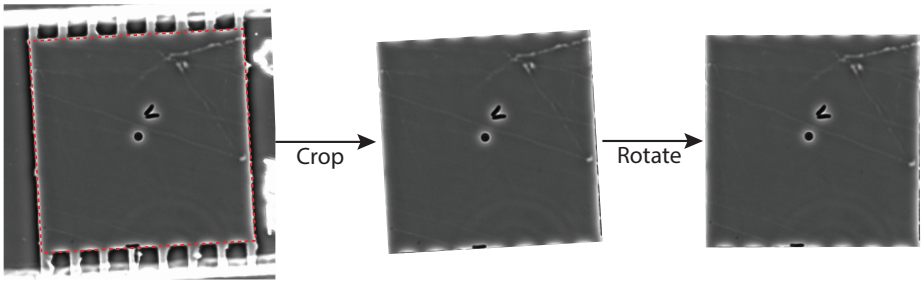


Figure 3.2: Cultivation chamber with the growth site boundaries as the red, dashed rectangle. The image is cropped to this region and subsequently rotated.

Several approaches to finding rectangular structures in gray scale images have been proposed in literature. Most of them are based on a transformation of the image into a domain, where prominent features of lines or rectangles are emphasized. Those approaches can be put into two categories based on their input.

If it is possible to describe the structure of interest in a functional way (i.e., an analytical expression is known), a Hough transform [77, p. 755-760] is well-suited. The basic concept of the Hough transform is to transfer points from the xy-plane to a

parameter space based on the analytical expression of the geometry of interest. The concept will be shortly explained on the example of finding lines in a binary image in the following.

Considering a point (x_i, y_i) in the x, y-plane, infinitely many lines $y_i = ax_i + b$ pass through that point. Rewriting the line equation as $b = -x_i a + y_i$ results in a single line in the a, b-plane for the fixed point (x_i, y_i) (cf. Figure 3.3). When adding an additional point (x_j, y_j) to the image, the a, b-plane will have a second line for all the lines passing through (x_j, y_j). If the two lines are intersecting, the point of intersection (a', b') denotes a parameter configuration and, thus, a line that passes through both points (x_i, y_i) and (x_j, y_j).

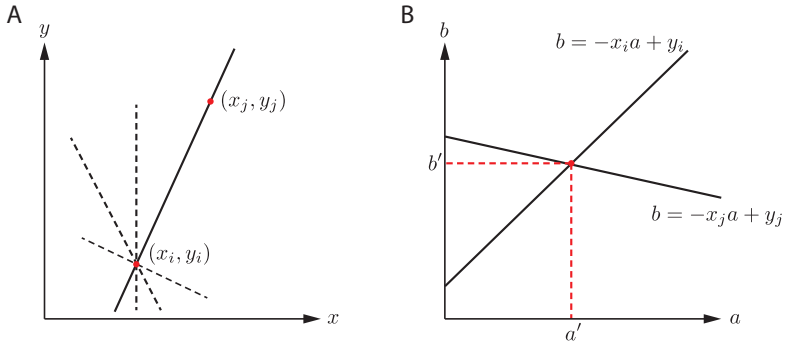


Figure 3.3: Transformation from xy- to ab-plane. **A.** A line passes through two points (x_i, y_i) and (x_j, y_j). Dashed lines are some examples for lines that pass through (x_i, y_i). **B.** (x_i, y_i) and (x_j, y_j) in x, y-plane are lines in a, b-plane. The connecting line is reduced to one point (a', b').

An algorithmic approach to finding lines in a binary image would thus be, to look for points in the parameter space where many lines are crossing. In the same spirit, a Generalized Hough Transform (GHT) [78] might be used for finding rectangles in an image. Furthermore, it is possible to explore geometric characteristics of a rectangle in the domain of the Hough transform and use such characteristics for the rectangle detection directly in Hough space [79].

The second category of approaches are template matching approaches [80]. Here, the general concept is to find instances of a provided, abstract template in the (unprocessed) image. To that end, a template is constructed that highlights prominent features of the structures one is looking for. When convolving the original image with the template, pixels with high values will denote regions where the overlap between the image and the template is high. Such approaches are, however, problematic for complete images of MGCs because of production uncertainties and their sensitivity to noise.

We have, hence, decided to stick with a simple but robust method that is based on proven theory and is available as an optimized implementation. This approach is presented in the following section.

3.2.1 Realization of Growth Site Detection

This method combines a custom edge detection technique (can be replaced with a Canny edge detection [81]) that generates a binary, edge-focused version of an image, with a traditional Hough transform to find lines in the image. Combining this approach with a heuristic to find matching pairs of parallel lines that connect to rectangles of the size of previously know chip structures determines the ROI for further processing steps.

Data: Image I
Result: Image cropped to growth site $I_{cropped}$
Detect edges in I;
Apply Hough transform and store n lines;
Find lines that are on opposite sites of the center of the image in horizontal and vertical direction, respectively;
while *lines available* **do**
  while *lines on opposite sites available* **do**
    if *distance is known distance and orientation is the same* **then**
      store pair of lines;
    else
      continue;
    end
  end
end
Crop image to the found rectangle;

Algorithm 2: Pseudo code of box detection.

The edge detection is based on templates that are characteristic for the edges of a MGC in a bright-field image. The general concept is to convolve the input image with the template images (cf. Figure 3.4). This convolution will assign each pixel in the input image a value that denotes how much the template and the surrounding of a pixel overlap. When there is more than $90\,\%$ overlap, a pixel will be marked as belonging to one of the edges of a growth site.

In the last step, the cropped ROI is rotated such that the top and bottom edges are horizontally aligned. This rotation step is implemented to avoid artifacts along the edges of the ROI in the remaining processing steps.

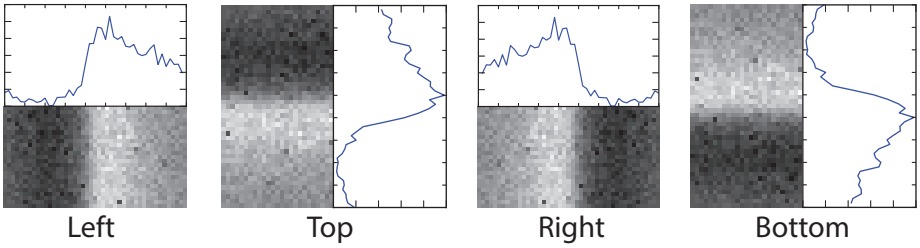


Figure 3.4: Reference images extracted from the four sides of a growth site in a phase-contrast image of an MGC. References are used for the detection of growth site boundaries and the subsequent cropping. Intensity profiles are presented next to the reference image.

3.2.2 Conclusions

While the proposed method for growth site detection is robust, in its current form it is only applicable for rectangular MGCs. Due to the optimized form of edge detection with pre-built references, finding the growth sites is still feasible with a brute force approach. More sophisticated methods (i.e., not limited to lines) on the other hand allow for a more general approach to growth site detection.

Template matching methods, e.g., the GHT, provide a general framework to find instances of provided templates in an image. Although we have to provide a template, it does not necessarily have to show a rectangular region. In a template matching framework it would be possible to detect round growth chambers (see [59]). A further upside is, that templates are readily available due to the fact that the LOC devices are custom designed and produced at the Helmholtz Nanoelectronic Facility (HNF) of the Forschungszentrum Jülich. Hence, schematics of cultivation chambers can be used after some conversions as input for such algorithms, provided CAD files are available. Nevertheless, the computational demand of general template matching approaches is immense even for optimized approaches. For a comprehensive assessment of computational aspects of template matching see [80, p. 201-11].

4 Cell Detection

In order to be able to extract and quantify information about individual cells from microscopic images, cells and their contours have to be detected. While it might be enough for some applications, to determine the absolute cell number in an image, we are in most applications interested in cellular quantities such as cell length, cell area, and fluorescence reporter intensities. Cell contours contain additional information when studying microbial growth, but especially, when working with fluorescence reporters they are mandatory. In such applications the main goal is to extract mean fluorescences over an individual cell area. As an example, Figure 4.1 shows two rather dense populations of *C. glutamicum* cells. The goal of the cell detection pipeline is to detect individual cells in entire image sequences.

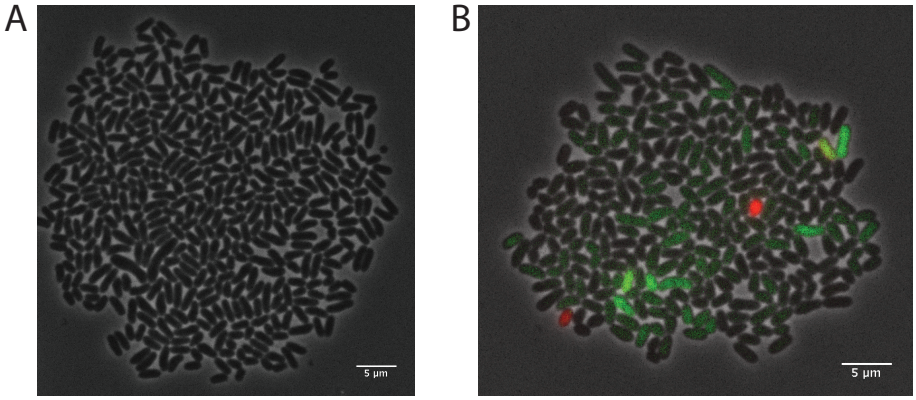


Figure 4.1: Dense populations of *C. glutamicum*. **A.** Phase contrast image only. **B.** Composite image of one phase contrast and two fluorescence channels.

The major challenges for segmentation in such images are

1. touching cells, where there is little to no gray value difference between the boundaries of two cells;

2. high frequency noise in the background (i.e., growth site regions without cells);
3. uneven intensity distributions throughout populations, such that cells near the center appear to be lighter compared to cells at the perimeter.

In the following sections, different approaches for the segmentation of images, which have various disadvantages for the acquired image data, are highlighted. Finally, the realization of the cell detection pipeline that has been developed in this PhD project is presented with the extension of a high-quality segmentation for post-processing.

4.1 Histogram-based Approaches

The most simple approach (after manual segmentation) is the concept of considering intensity values of each pixel in an image. Most of these approaches are working on the so called gray value histograms and do, therefore, not take into account the spatial location of a pixel. Due to the independence of the spatial location these approaches do not preserve object integrity. This means, that pixels that are spatially located in proximity might be on two opposites in the domain of the gray value intensities.

4.1.1 Thresholding

Thresholding is a generic concept that describes the separation of foreground and background based on a threshold in a gray value histogram. Various approaches to the separation problem are available [82]. Some are based on the shape of the histogram and use curvature, others apply classification methods to find clusters that correspond to the image regions. Figure 4.2 shows an example of a gray value histogram as well as the idea of thresholding. A thorough comparison of available thresholding techniques is given in Sezgin and Sankur's survey on the topic [82].

After the computation of a threshold, each pixel's gray value is compared to a computed threshold and based on that either classified as foreground or background. Therefore, the resulting output image will be of binary format as computed by

$$o(x,y) = \begin{cases} 0 & \text{if } I(x,y) \leq \theta \\ 1 & \text{else} \end{cases} \tag{4.1}$$

where $I(x,y)$ denotes the gray value of a pixel (x,y).

Techniques differ in the computation of the threshold θ, especially which region of the image, i.e., the whole image or just a window is considered. While global thresholding techniques, e.g., Otsu's method [83], work on the histogram of a complete image, local adaptive thresholding techniques, e.g., Sauvola's method [84], compute a local threshold $\theta(x,y)$ for each pixel.

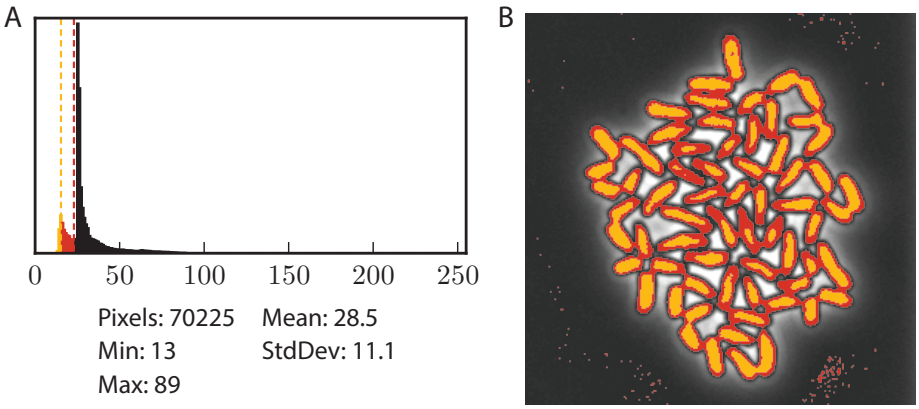


Figure 4.2: Histogram-based approaches. **A.** The grey value histogram of the image shown in B. Two different thresholds are selected (orange and red). **B.** A composite image of the input and the resulting segmentations from the two selected thresholds in A. Pixels in orange and red, respectively, are classified as foreground. All other pixels are background.

Global Thresholding. The method proposed by Otsu in 1979 [83] is based on interpreting the histogram of an image as probability density distribution. To that end, the occurrences of gray values (i.e., the height of bins) have to be converted to probabilities by normalizing to the total number of pixels of the image. The separation procedure is based on the assumption that the image consists of two classes, foreground and background. Therefore, the final threshold is selected to separate those two classes as well as possible. In mathematical terms, the within-class variance (σ_W^2) is subject to minimization:

$$\sigma_W^2(\theta) = \omega_1(\theta)\sigma_1^2(\theta) + \omega_2(\theta)\sigma_2^2(\theta),\ \theta \in [0, 255] \text{ for 8-bit images} \tag{4.2}$$

with weighting factors

$$\omega_1(\theta) = \sum_{i=0}^{\theta} p(i) \qquad \text{and} \qquad \omega_2(\theta) = \sum_{i=\theta+1}^{255} p(i).$$

In the above formulations, $p(i)$ denotes the probability of gray value i, and σ_1^2 and σ_2^2 the threshold-dependent variances of the two classes, foreground and background, respectively. The basic idea is to find a constant threshold θ that minimizes the within-class variance. In an iterative manner, the variance for each value of θ is computed in order to determine the optimal global threshold.

The evaluation of Equation 4.2 involves the computation of variances σ_1^2 and σ_2^2. The relationship between the within-class variance and the between-class variance

(σ_B^2), however, can be facilitated to ease the computational effort. Equation 4.3 shows the aforementioned relation. With σ being the standard deviation of the complete histogram and, thus, being a constant factor, the inter-class variance is maximized in order to find the optimal threshold θ. This only involves computation of mean values μ_1 and μ_2 instead of standard deviations (second order).

$$\sigma_B^2(\theta) = \sigma^2 - \sigma_W^2(\theta) = \omega_1(\theta)\omega_2(\theta)[\mu_1(\theta) - \mu_2(\theta)]^2 \tag{4.3}$$

with mean values

$$\mu_1(\theta) = \sum_{i=0}^{\theta} p(i) \cdot i \qquad \text{and} \qquad \mu_2(\theta) = \sum_{i=\theta+1}^{255} p(i) \cdot i \tag{4.4}$$

In mathematical terms, the optimal threshold is defined as

$$\hat{\theta} = \underset{1 \le \theta \le 255}{\operatorname{argmax}} \, \sigma_B^2(\theta).$$

Local Adaptive Thresholding. Sauvola's method, reported in [84], uses information from a $w \times w$ pixel window of the original image to determine a location-dependent threshold $\theta(x, y)$ for each pixel $(x, y) \in \Omega$. Using localized information from a window instead of a global histogram avoids effects of uneven gray values of foreground and background in different regions of an image. The threshold $\theta(x, y)$ is computed according to

$$\theta(x, y) = \mu(x, y) \left[1 + k \left(\frac{\sigma(x, y)}{R} - 1 \right) \right] \tag{4.5}$$

where R is the theoretical maximum value of the standard deviation, i.e., in case of an 8-bit gray-scale image $R = 128$. Additionally, the first-order statistic of the local mean values, $\mu(x, y)$, is incorporated, as well as k as a weighting factor. The calculation of $\sigma(x, y)$ as well as $\mu(x, y)$ are based on the $w \times w$ window centered at (x, y).

Phansalkar *et al.* [85] have proposed a refined method of Sauvola's thesholding, with an additional (exponential) weighting factor

$$\theta(x, y) = \mu(x, y) \left[1 + p \cdot \exp(-q \cdot \mu(x, y)) + k \cdot (\frac{\sigma(x, y)}{r} - 1) \right] \tag{4.6}$$

Bernsen [86] proposes a local gray range technique. In this technique the range between the maximum $\max(x, y)$ and minimum $\min(x, y)$ gray value within a local window centered at (x, y) is used to determine the threshold value

$$\theta(x, y) = \begin{cases} 0.5(\max(x, y) + \min(x, y)) & \text{if } C(x, y) \ge 15 \\ 0 & \text{else} \end{cases} \tag{4.7}$$

with the local contrast $C(x, y) = \max(x, y) - \min(x, y)$.

Thresholding works particularly good if foreground and background are separated well in the location-independent gray value domain. That is, cells have a more or less constant gray value, whereas the background has a significantly deviating gray value. In this case, the two regions are separated by setting a value in the histogram such that pixels with a value above the threshold are assigned to one region and the other way around for pixels below the threshold (see Figure 4.2).

Especially for our acquired images, where the mean gray value of cells is location dependent, global thresholding techniques are not applicable. This effect is visualized in Figure 4.2 where the yellow threshold covers a significantly reduced portion of cell area for cells near the center of the population. In addition, segmenting touching cells while separating them from the background is barely possible, when the detected cell area is of interest.

4.2 Non-histogram-based Approaches

Histogram-based approaches utilize gray values of pixels irrespective of their spatial location. Taking spatial location into consideration allows for using neighborhood information of pixels. The algorithms that fall in this category will be discussed in the following paragraphs. General advantages as well as disadvantages will be highlighted when the algorithms are applied to the images that are acquired using the setup described in Section 2.3.

Watershed Segmentation. Another segmentation strategy is the popular Watershed transform [87, 88]. The general concept is to interpret gray value intensities of an image as the height for each pixel. Hence, the image can be interpreted as a topographic relief map with basins and watersheds. To separate the basins, the concept is to flood the relief over time (in an iterative procedure) and observe when sinks are merged together. When two basins are merged, a ridge can be extracted from the previous iteration.

The downside of this approach is its sensitivity to changes in gray value. Hence, images have to be typically smoothed before the Watershed transform can be applied. Otherwise, it tends to heavily oversegment regions of almost constant gray value. Figure 4.3 shows the Watershed transform applied to an example image with and without smoothing.

Template Matching. A more sophisticated method that does not only take into account the pixel information of images are so called *template matching* approaches. The general idea of template matching is to find a predefined template or set of templates (i.e., in our case cell images) in the input image for the algorithm [80]. To that end, a

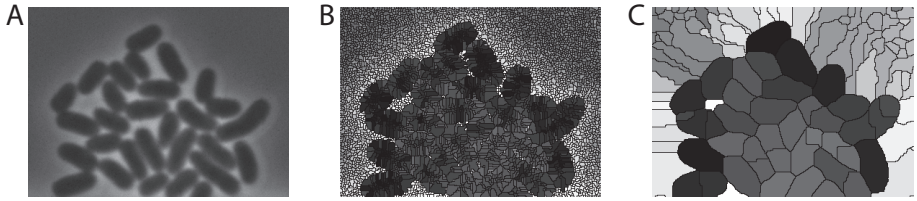


Figure 4.3: Watershed segmentation applied to image data. **A.** Input image of a *C. glutamicum* population. **B.** Watershed transform applied to the input image showing the issue of oversegmentation. **C.** Watershed transform applied to a smoothed version of the input image.

multitude of templates has to be produced for the different types of objects. However, this process can be eased by parameterizing the templates and checking for different parameter sets in the image.

These approaches are formulated as an optimization problem and can, hence, be solved using available optimization algorithms. A disadvantage for such approaches is the necessity for good templates that match the objects depicted in the input image. If only little information is known about the objects in the image, i.e., objects can occur with any rotation angle and any scaling parameter, template matching approaches are computationally very demanding.

Furthermore, template matching approaches will barely result in good detections if the quality of the template is low. While for most of cell types, it is possible to come up with an appropriate template, this is rather restricting with templates that feature high degrees of freedom, e.g. rods. With each possible degree of freedom in the template, the search space for the optimal fit of the template in the acquired image is extended by a new dimension.

Active Contours. An even more sophisticated approach to image segmentation using deformable models has been published almost 30 years ago but has been refined in recent years: active contours [89]. Active contours or snakes are a very popular approach in this category, though rather computationally intensive. A snake is a closed curve in two-dimensional space that is optimized to split the image domain into object and outside of an object along the object's boundary. In an iterative procedure, an initial curve is set (either manually or by a heuristic) that is evolved in the image domain either in outward direction or to the inside. During the iteration procedure the snake evolves like a wave front. The curve converges to a minimal energy that is defined by the user, usually referred to as snake energy. In our case, the energy is defined in a way that the curve reaches the minimal energy when it is on the contour of a cell.

An advantage of using active contours is the interactivity. Users can select the initialization of the iterative evolution of the snake and, furthermore, can select points

of interest that should be parts of the converged snake. That said, active contour approaches are very flexible but their performance heavily relies on a good initialization. For this reason, we have decided to discard active contours as a putative method for the automated segmentation of entire images. However, we have implemented a specialized snake representation and energy formulation for refinement cells that are hard to split with the realized segmentation procedure. Details on this will be provided in Section 4.7.

Machine Learning Approaches. Machine learning approaches use specified feature vectors for pixels that are used for classification. The approaches fall into two categories. Unsupervised approaches, will try to classify pixels only based on each pixels feature vector. In contrast, supervised approaches have to be trained on data. Hence, the input to supervised approaches is a combination of feature vectors of pixels and the desired classification.

One implementation of a supervised classification approach is a Support Vector Machine (SVM). An SVM is trained on a set of data for which classifications of individual pixels are available (the so called training step). The trained model is subsequently used for the classification of images that share similar characteristics as the image (set) the model was trained on.

The feature vector for each pixel can for example contain several gray values from different representations of an image [90]. A possible representation that contains additional information is the gradient image or the second-order derivative image. Such representations span an additional axis in the feature space for each individual pixel. Using the feature vectors of known samples, an SVM determines a separating (hyper)plane in the multidimensional feature space such that the gap between the data points and the (hyper)plane is maximized.

In contrast, unsupervised clustering techniques aim at clustering pixels in an image into K clusters, without being previously trained. To that end, algorithms are initialized with K clusters. Each pixel in an image is assigned to the cluster that is closest. Upon adding the pixel to this cluster, the center of said cluster is recomputed and the computation is continued with another pixel. The definition of closeness in such algorithms can incorporate prior information. Popular metrics are the spatial location in the image domain as well as the gray value. Weighted combinations of the aforementioned characteristics are also possible, and even advisable.

K-means clustering and its fuzzy equivalent, fuzzy c-means (FCM) has proven to be a solid approach to detecting clusters in various fields of computer science and image processing in particular [91]. However, it lacks the possibility to deviate from the initial number of clusters. Although heuristics for finding good values for K are available [92], they have to be optimized for specific use cases.

4.3 Realization of Cell Detection

Although sophisticated segmentation approaches are available (see previous section), they are in many cases not performing reliably for our use cases. Due to the specific nature of images that are to be processed with the image analysis pipeline, we have implemented an optimized segmentation procedure for dense populations of *C. glutamicum*. A general overview of the pipeline with all major steps is presented in Figure 4.4.

Arguably the most complex step of the pipeline is the computation of the so-called Shape Index Map (SIM). The SIM is a curvature descriptor in the spatial domain that is computed for each pixel in an input image I (Figure 4.4-(0)) resulting in I_{sim} (Figure 4.4-(1)). More details on the computation of the SIM are presented in Section 4.4. The following sections will provide a general overview of the steps involved in the realized pipeline.

First, the input image is blurred with a Gaussian kernel (step omitted in Figure 4.4) because methods based on the gray value gradient or higher order derivatives are sensitive to noise. Parameters for the smoothing kernel depend on the acquisition settings. Here, the major influencing factor is the signal-to-noise ratio (SNR) of raw images. From experience, we have set the default size of the Gaussian kernel to $5\,\mathrm{px} \times 5\,\mathrm{px}$.

Subsequently, the SIM is globally thresholded. The selection of an appropriate threshold depends on the specific application and the investigated organism. The SIM is constructed such that values near -0.25 denote regions where cells are touching. The output of this thresholding will henceforth be called $I_{thresholded\_sim}$ (Figure 4.4-(2)).

Next, a median filter is applied to $I_{thresholded\_sim}$. In general, a median filter replaces the center pixel of the processing window with the median of the gray values in the window. Median filters are quite popular for image processing tasks as they provide excellent noise reduction capabilities for certain types of noise [77, p. 178]. In practice, applying a median filter to $I_{thresholded\_sim}$ removes smaller holes in cellular structures that are due to not perfectly constant gray values within a cell, creating $I_{cleaned\_sim}$ (Figure 4.4-(3)). Additionally, the filter also removes salt-and-pepper artifacts (i.e., high frequency noise) in regions of the growth site where no cells are located. Parameters for the median filter are set in accordance with the resolution of the camera that is used for image acquisition. While the cameras that have been used in the beginning of the project have a spatial resolution of about $7\,\mathrm{px/\mu m}$, newer cameras have a 2-fold increased resolution of $15\,\mathrm{px/\mu m}$. Hence, the width of the filter kernel is adapted to be approximately thrice the width of an individual *C. glutamicum* cell, i.e., $20\,\mathrm{px} \times 20\,\mathrm{px}$ and $40\,\mathrm{px} \times 40\,\mathrm{px}$, respectively. Figure A.1 shows enlarged versions of the postprocessing steps of the SIM. The aforementioned artifacts will be removed in a second branch of the processing procedure, described in the following.

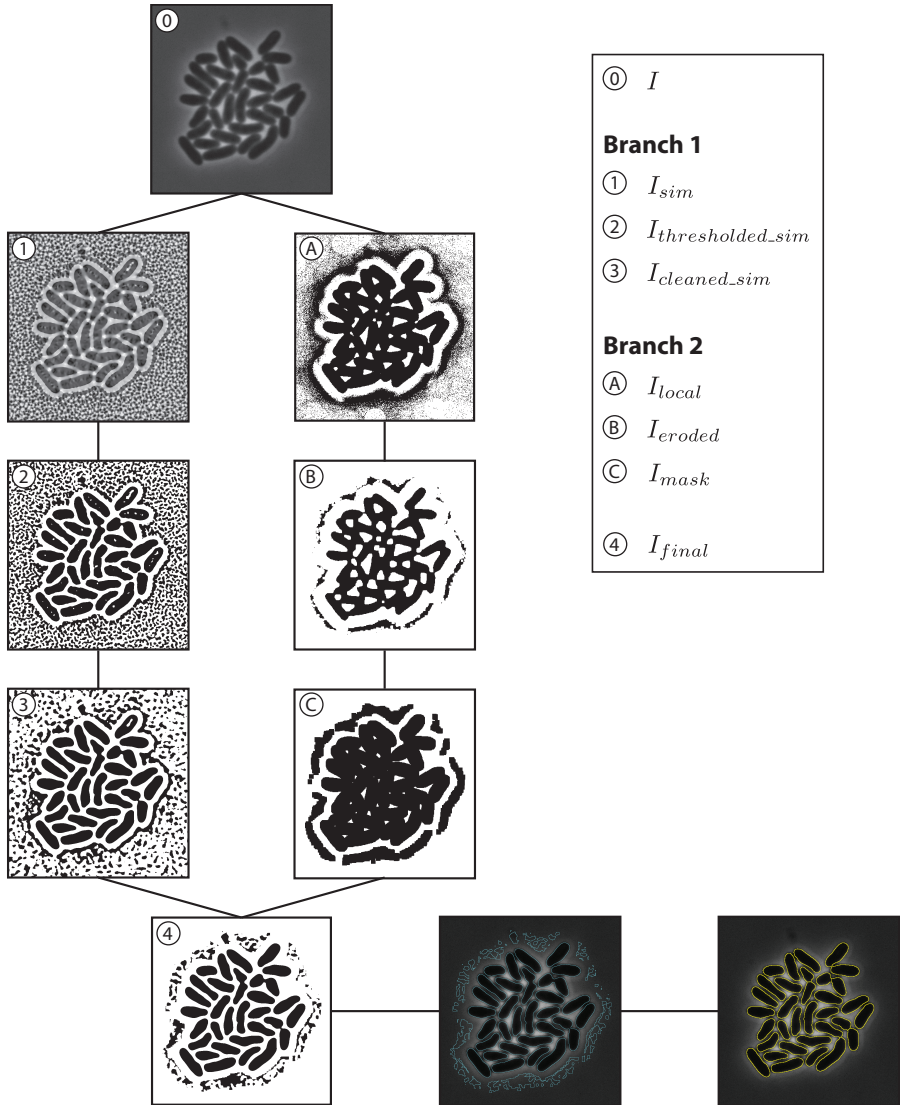


Figure 4.4: A general overview of the cell detection procedure that is applied in the image analysis pipeline. Starting from a preprocessed input image I, two parallel branches create a segmentation of a population and a mask for the population, respectively. Results from the branches are combined to create I_{final}. The resulting segments are filtered and clustered cells are further processed. Detailed information is provided in the text.

In a first step, I is locally thresholded with one of the previously described local thresholding method (cf. Section 4.1.1, resulting in I_{local} (Figure 4.4-Ⓐ). The default local thresholding approach is Sauvola's method [84]. Please mind that the input to the local thresholding is the original image I, not $I_{thresholded\_sim}$. Parameters are again subject to changes and optimization for the experimental setup.

In order to remove salt-and-pepper noise that is prevalent in regions surrounding the population, multiple erosions are applied (I_{eroded}, Figure 4.4-Ⓑ). A morphological erosion effectively removes objects from the image, that are smaller than the kernel. By repeatedly applying an erosion, larger objects will basically be slimmed down to the size of the kernel along their boundaries. Hence, multiple dilations will in practice enlarge the population's cells such that a mask image for the population is generated (I_{mask}, Figure 4.4-Ⓒ). One can see in the images in the overview that the population is surrounded by a band of artifacts. These artifacts could have been removed by more erosions. However, this poses the risk of disappearing cells, rendering the subsequent dilations useless. Detailed images of the outputs of the previously described branch of the cell detection scheme are shown in the appendix (cf. Figure A.2).

To generate a binary image I_{final}, $I_{thresholded\_sim}$ and I_{mask} are combined (Figure 4.4-④). This is achieved by applying an **OR** operation to each pixel coordinate in the two thresholded, binary images (convention: white pixels $= 1$, black pixels $= 0$). The concept behind the pixel-wise **OR** operation is, that in the outer regions of the population the thresholded image I_{mask} is 1 so that those region are effectively cleaned in the output image (no matter what the value is in $I_{thresholded\_sim}$). Since the population region in I_{local} is 0, the values from the processed SIM image $I_{thresholded\_sim}$ are pivotal.

This step effectively creates a binary image, where foreground and background are separated. Subsequently, a labeling algorithm scans the image in rows until a foreground pixel is found. Starting from that pixel, the contour of the region is traced [77, p. 818-20]. The contour tracing is based on a robust 8-connected neighborhood of each pixel. Details on how the SIM is used for detecting individuals cells in dense populations are provided in the following section.

4.4 Processing Dense Populations with the Shape Index Map

Dense bacterial populations are characterized by a high number of cells in a defined volume. As a result, cells are touching with little change in gray value along the cell boundary. Figure 4.5 shows an exemplary population of four cells in a three-dimensional representation, where the z-axis denotes the gray value.

Instead of only using the image gradient (first derivative) the principal curvatures of the Hessian can be used for characterizing the connection points of cells. A measure

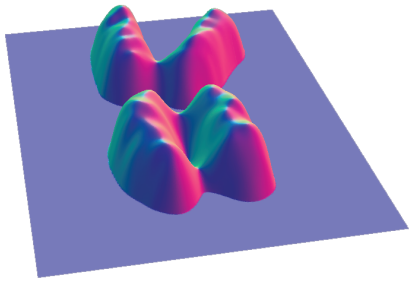


Figure 4.5: Image of a microbial population, where the gray value intensity is shown in z-direction. The color codes for the gradient, where pink and cyan denote opposite ends of the spectrum.

based on the principal curvature is the SIM as introduced by Robbin and Salmon [93] in their publication on shape theory.

The differential structure of a surface is captured by the local Hessian matrix, which is approximated in terms of surface normals by

$$\mathcal{H} = \begin{pmatrix} -(\frac{\partial n}{\partial x})_x & -(\frac{\partial n}{\partial x})_y \\ -(\frac{\partial n}{\partial y})_x & -(\frac{\partial n}{\partial y})_y \end{pmatrix} \tag{4.8}$$

where $(\cdots)_x$ and $(\cdots)_y$ denote the x and y components, respectively. The principal curvatures of the surface are the eigenvalues of the Hessian matrix, found by solving $|\mathcal{H} - \kappa \mathbf{I}| = 0$ for κ, where $\mathbf{I}$ is the identity matrix. Koenderink and van Doorn [94] developed a single-value, angular measure to describe local surface topology in terms of the principal curvatures. This shape index is defined as

$$s = \frac{2}{\pi} \cdot arctan \left(\frac{(\frac{\partial n}{\partial x})_x + (\frac{\partial n}{\partial y})_y}{\sqrt{\left((\frac{\partial n}{\partial x})_x - (\frac{\partial n}{\partial y})_y\right)^2 + 4(\frac{\partial n}{\partial y})_x(\frac{\partial n}{\partial x})_y}} \right) \tag{4.9}$$

in terms of surface normals. The partial derivatives are approximated by first-order differences. Examples for the measure are shown in Figure 4.6.

4.5 Detecting Cell Clusters

After application of the previously described pipeline it is possible that multiple cells are assigned one label (i.e., individual cells have not properly been detected). In this case, we refrain from further massaging the available image information and instead incorporate additional information about the objects we seek to detect. In case of

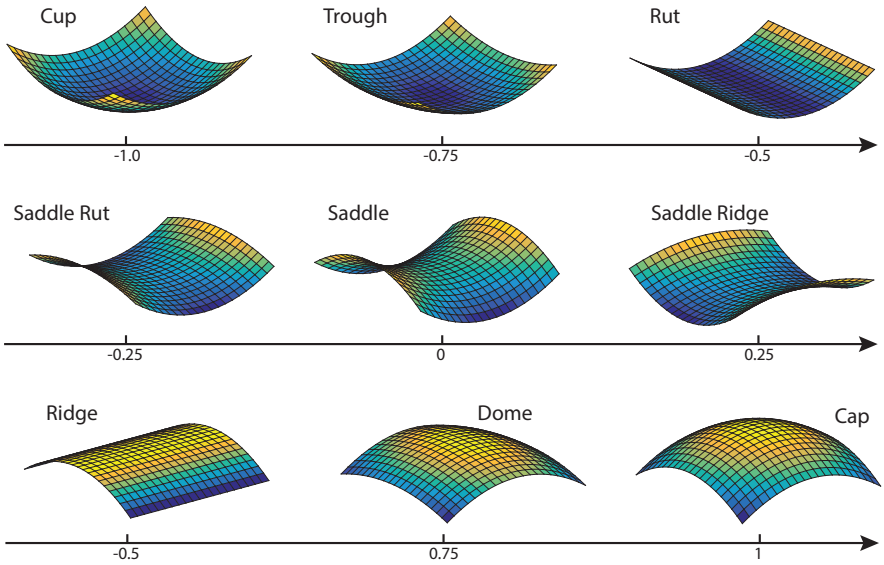


Figure 4.6: Examples of the nine categories into which the values of the SIM fall (adapted from [94]).

C. glutamicum we are looking for rod- to ellipse-shaped bacteria that form characteristic clusters when close to each other.

However, before we are able to further process such clustered cells it is necessary to detect those cases. To that end we have analyzed cells to find cellular properties that help discriminate individual cells from cell clusters. Additionally, this step is crucial to reduce the computational load of the pipeline because it reduces the amount of cells that have to be processed in order to generate a correct segmentation. In general, this is implemented as an iterative procedure, where a filtering step is proceeded by a splitting step and so forth (with an upper limit for the number of iterations). The procedure stops because clusters are split up into individual cells leaving only individual cells after some time. The implemented filters are presented in the following sections as well as some thoughts on how these methods could be further improved or extended.

4.5.1 Cell Size Filter

A rather simple but robust approach is a size filter. The concept is to disregard objects that are too large to be a cell cluster of more than ten times the average cell area and smaller than one tenth of the average cell area. Objects that fall in the range between the two boundaries are categorized into single cells when their size is below a given threshold and multiple cells when their size is above a provided threshold

(cf. Figure 4.7). Further processing is done on objects that fall into the multiple cells category.

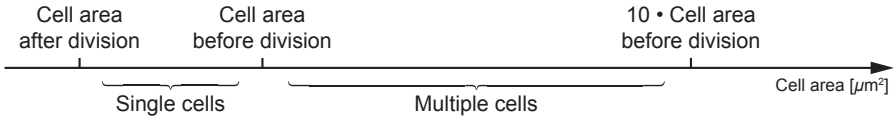


Figure 4.7: Different thresholds of the size filter explained.

4.5.2 Convex Hull Filter

The concept of the convex hull filter is to compute the convex hull of an object and compute the ratio of its area to the convex hull as shown in Figure 4.8. This ratio is also called solidity in literature. The deviation from a ratio of 1 should be set to about 0.20. The lower the value is set, the more perfectly shaped a single cell has to be to not be split.

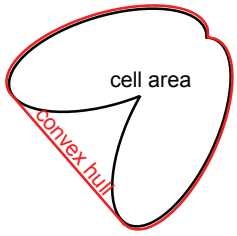


Figure 4.8: Two artificial cells (motivated by the V-snapping of *C. glutamicum*) that are detected as one object and, therefore, have only one outer contour. The red line shows a schematic of the convex hull of the contour.

4.5.3 Size and Convex Hull Filter Combined

The idea is to filter small and large objects according to the minimal cell and maximal cell size, respectively (see Figure 4.7). Consequently, the solidity filter is applied to the remaining objects.

4.6 Splitting Cell Clusters

All aforementioned segmentation methods produce unsatisfying results for overlapping cells. Although overlaps of cells are minimized by the structure of the cultivation chambers in the used LOC devices (see Section 2.2), such issues are frequently observed.

Bai *et al.* [95] propose a method exploiting the convexity of cells. Hence, the approach is only applicable to cells that are convex. Touching cells, however, do rarely generate convex objects but have regions of concavity. This observation is used for the computation of putative splitting points. Figure 4.9 shows two cells with the proposed splitting points. Algorithmically, the splitting points are determined by computing an approximation of the contour of a cell cluster as a polygon. Consequently, normal vectors to the approximated contour are constructed.

In the next step, the angle between two adjoint vectors is calculated. If an angle that exceeds a given threshold is identified, a splitting point is proposed, as shown in Figure 4.9. Subsequently, the points are connected with a line that separates the two cells.

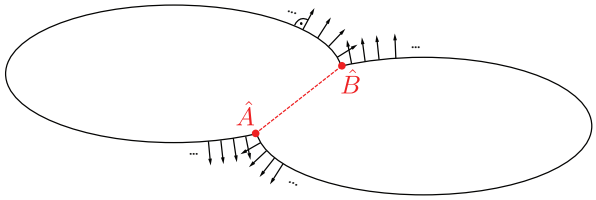


Figure 4.9: Two cells that have divided but still appear overlapping. The black lines, orthogonal to the contour, denote the outward directed normal vectors. The red points mark the proposed splitting points that are connected via the red dashed line which is the splitting result.

Wang *et al.* [96] have proposed an approach that finds the best fitting pair of splitting points for overlapping cells by computing an energy for each pair of points of the contour. So-called bottleneck positions are found by minimizing the energy function

$$E_S(A,B) = \frac{\|A-B\|_2}{\min\big(\text{length}(A \to B), \text{length}(B \to A)\big)} \tag{4.10}$$

where A and B are the points for which the energy is to be computed. Let $\bar{\mathcal{R}}$ be the set of points on the contour. Furthermore, length$(X \to Y)$ denotes the length of the contour segment, walking from X to Y on the contour. Vice versa, length$(Y \to X)$ is the distance that has to be walked along the contour to reach X, starting from Y.

Finally, the two splitting points $\hat{A}$ and $\hat{B}$ are determined by solving the optimization problem:

$$(\hat{A},\hat{B}) = \underset{A,B\in\bar{\mathcal{R}}}{\operatorname{argmin}}\, E_S(A,B). \tag{4.11}$$

Figure 4.9 illustrates that at the two proposed splitting points $\hat{A}$, $\hat{B}$ the distance between the points is minimal and the length of the two contour segments is approximately equal.

4.7 Incorporating Additional Knowledge into the Segmentation Process

If knowledge about the image composition (i.e., the objects in an image) is available, segmentation procedures can integrate that information for improving results. The splitting approach that has been presented in the previous chapter already takes into account information about the the morphology of bacteria for finding putative splitting points. This is, however, not done during segmentation but is established as a post-processing procedure for cell clusters.

In her Master's thesis, Karin Bokelmann has implemented a cell segmentation procedure based on a deformable model, i.e., parametric snakes. Such models can be used to describe cellular properties and, hence, incorporate prior knowledge in to the segmentation process. The snake energy that determines the convergence of the snake is a functional of the snake representation itself. Hence, the objective is to find the best snake (i.e., a function) that minimizes the snake energy. The segmentation is embedded in a variational framework [97] because of the functional nature of the snake energy (cf. Figure 4.10 for example). The adaption and implementation of an energy formulation for ellipse-reproduction and the extension to rods will shortly be described in the following. For more details, the reader is referred to Karin Bokelmann's thesis [98].

While it is in theory possible to use this method for stand-alone segmentation, the computational demand for segmenting a complete image of a growth site is immense. This is further emphasized by the fact that snakes have to be sensibly initialized to generate results in reasonable time. In cases, however, where a user initializes a snake for an individual cell, the approach successfully reconstructs cell contours with subpixel resolution. Therefore, we use the below described method as a post-processing step where the previously describe cluster splitting has failed. In such cases, the user manually removes the splitting result and initializes a snake for a cell. The snake will converge to the cell contour, when a reasonable parameter set is used.

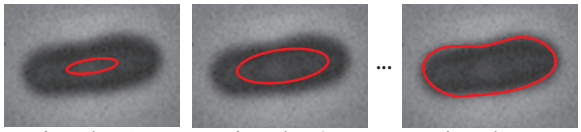


Figure 4.10: Snake evolving to the final contour of the cell. The snake has been initialized with an ellipse that is extended in direction of the outward normal to reach convergence at the final state.

4.7.1 Active Contours and Snakes

Active contours and, in particular, snakes are effective tools for the segmentation of images. The multitude of snake variants differ in their type of curve representation and in the formulation of energy terms [89, 99–104]. With regard to the curve representation, snakes can be categorized into three major groups:

1. point snakes,
2. parametric snakes, and
3. implicit snakes.

For a point snake, the curve is solely represented with points along the contour. This represents a special case of a parametric snake with many control points. While the curve of a parametric snake is represented continuously by coefficients of a set of basis functions at control points on the contour, implicit snakes rather evolve a 2-D surface. In the latter cases, the curve is implicitly described by the "level set" of a surface. A level set in that particular case is mathematical function $\phi(x, y)$, that assigns to each coordinate $(x, y) \in \Omega$ a value such that:

$$\phi(x,y) = \begin{cases} < 0 & \text{if } (x,y) \in \mathcal{R} \\ 0 & \text{if } (x,y) \in \bar{\mathcal{R}} \\ > 0 & \text{if } (x,y) \notin \mathcal{R} \cup \bar{\mathcal{R}} \end{cases} \tag{4.12}$$

where $\mathcal{R}$ denotes the region that is defined by the contour $\bar{\mathcal{R}}$.

Snake Representation. A closed curve $r(t)$ in the plane is described by the Cartesian coordinate functions $x(t)$ and $y(t)$, where $t \in \mathbb{R}$ is an artificial, continuous parameter. The two functions x and y are parameterized by linear combinations of suitable basis functions φ. When looking at closed curves that are specified by a sequence of M control points $\{c[k]\}_{k\in\mathbb{Z}}$ (M-periodic), the following vectorial equation describes the curve (e.g., Figure 4.11):

$$r(t) = \sum_{k=-\infty}^{\infty} c[k]\varphi(Mt - k). \tag{4.13}$$

Delgado-Gonzalo *et al.* [97] propose a minimum-support, ellipse-reproducing basis that is based on exponential B-splines:

$$\varphi(t) = \begin{cases} \frac{\cos\frac{2\pi|t|}{M}\cos\frac{\pi}{M} - \cos\frac{2\pi}{M}}{1-\cos\frac{2\pi}{M}}, & 0 \le |t| < \frac{1}{2} \\ \frac{1-\cos\frac{2\pi(\frac{3}{2}-|t|)}{M}}{2(1-\cos\frac{2\pi}{M})}, & \frac{1}{2} \le |t| < \frac{3}{2} \\ 0, & \frac{3}{2} \le |t| \end{cases} \quad , t \in \mathrm{R} \tag{4.14}$$

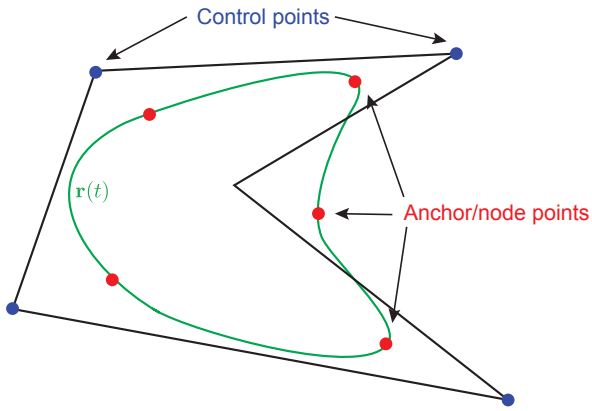


Figure 4.11: Parametric representation of a closed curve with control points and an artificial generating function.

The general idea of active contours approaches is to find a curve which resembles the boundary of an object in an image best, thus, segmenting the image. Starting with an initial guess for the wave (usually the image boundary), the evolution of this wave is driven by forces exerted by the image. Finally, the wave stops evolving when it resembles the object boundary and, therefore, has reached the minimal energy configuration. Hence, any active contour algorithm is driven by the chosen energy function.

Snake Energy. The proposed snake energy term comprises two factors, i.e., the influence of an edge as well as the region. Both terms are combined with a tradeoff parameter $\alpha \in [0, 1]$ that determines how balanced the contribution of the two terms is in the final formulation,

$$E_{snake} = \alpha E_{Edge} + (1 - \alpha) E_{Region}. \tag{4.15}$$

The edge energy E_{Edge} can be expressed as a surface integral using Green's theorem. The theorem gives the relationship between a line integral around a closed curve $r(t)$ and a double integral over the plane region $\mathcal{R}$ bounded by $r(t)$.

$$E_{Edge} = -\oint_r k^T (\nabla I(x_1, x_2) \times dx) \tag{4.16}$$

$$= -\int_{\mathcal{R}} \Delta I(x) dx_1 dx_2 \tag{4.17}$$

The region-based energy E_{Region} is built such that it discriminates an object from its surroundings, i.e. a defined shell around the snake. This is implemented by maximizing

the contrast between the intensity averaged within the region $\mathcal{R}$ and intensity of the data averaged over the elliptical shell $\mathcal{R}_\lambda$. The elliptical shell is constructed by finding the best fitting ellipse around the snake and scaling its axes by a factor λ. E_{Region} can thus be expressed as

$$E_{Region} = \frac{1}{|\mathcal{R}|}\left(\int_{\mathcal{R}} I(x)dx_1dx_2 - \int_{\mathcal{R}_\lambda \setminus \mathcal{R}} I(x)dx_1dx_2\right) \tag{4.18}$$

where $|\mathcal{R}|$ is a scaling factor that can be interpreted as the signed area.

4.8 Conclusions

While image segmentation is still a field of active research, the first approaches have already been presented in the 1950s [105]. It is commonly acknowledged that there is no one-fits-all approach that serves the specific needs (i.e., reliability, robustness, and accuracy) of biological applications [106, 107]. We have approached the presented challenges, i.e., dense populations with touching cells and uneven illumination, with a custom binarization pipeline and an advanced post-processing approach.

This pipeline consists of several modules that are executed serially. The first step is a heavily optimized binarization procedure, effectively separating cells from the image background. The resulting segments are filtered according to organism specific parameters, where clustered cells are detected. These clusters are treated by a custom splitting procedure that takes into account the morphology of individual cells. Finally, the user has the possibility to use the previously presented parametric snakes for segmentation of single cells where the splitting has failed.

All in all, the presented cell detection pipeline strikes a balance between organism specific features, flexibility, and robustness to the peculiarities of the acquired image sequences. We have developed and improved the cell detection module with great care due to its importance for the subsequent step in the pipeline: once cells are detected in each image of a sequence, correspondences of cells between subsequent images have to be found. Thus, the output of the cell detection procedure is used as the input for the tracking of cells.

5 Tracking of Cells in Image Sequences

The human brain is exceptionally good at identifying objects (that are visually perceivable) and recognizing their movement. There are, however, several factors where the tracking of cells in image sequences differs from the tracking of (immutable) objects with our eyes. Images are acquired at a much lower frame rate than the human perception and, more importantly, bacterial cells are not immutable objects. Bacteria are highly volatile cells that change drastically during a cell cycle. Between two subsequent images of an image sequence, cells can

1. change their volume and the observed cell area,
2. move in the MGC,
3. rotate,
4. split into two (or more) cells, and
5. leave the field of view.

In addition to tracking movements of individual cells, all these case have to be taken into account to reasonably capture the development of microbial populations in MSCC. Due to these challenges, algorithmic approaches are still under active development [108].

Given two (segmented) images of the same object, the process of finding the corresponding object in both images is referred to as "tracking". Figure 5.1 visualizes the problem with dots denoting centers of cells (output of the detection step is shown in yellow solid lines) at different time points t_{n-1} to t_{n+1}. The dashed lines show the potential assignments between cells between the subsequent images.

In general, given an image sequence of a single moving object, the goal is to reconstruct the trajectory of this object. If, however, several objects are present, the challenge is

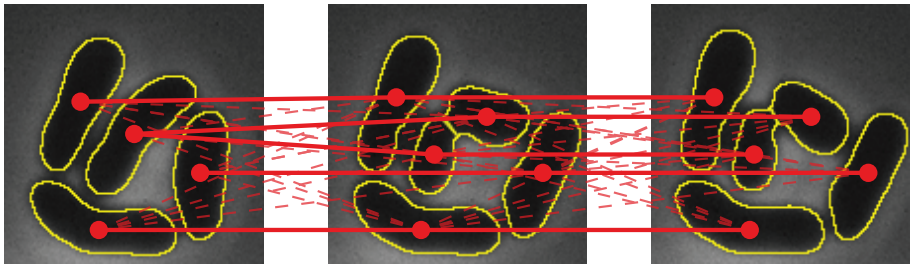


Figure 5.1: An image sequence where the yellow contours denote the detected cell areas, red dots are a simplification of the cells, and red lines are assignments between cells. Solid lines denote the correct assignment, dashed lines all the possible reconstructions.

to find corresponding objects throughout the image sequence. This task is referred to as "correspondence problem" or "assignment problem" in literature. For the simple example of Figure 5.1, optimal assignments have to be chosen from more than $10{,}000$ possible lineage reconstructions for three frames.

As microbial cells are only loosely attached to the LOC devices and are not restricted in the x, y-plane in growth sites, cells will move during an experiment. This is on the one hand owing to the constant flow in the chips, on the other hand due to cellular growth in a population (i.e., volume increase in a confined space). Because of this spatial movement, solving the correspondence problem in the acquired image sequence remains a challenging task. In addition to movements, cells are undergoing mitosis. Therefore, it is not enough to find one-to-one assignments between consecutive frames, but it is necessary to take one-to-two relationships into account.

Many approaches have been proposed and discussed in literature in recent years (see [108–110] for reviews). The presented approaches can be categorized according to their ansatz: tracking after detection and tracking with detection. The former approaches work with a separate cell detection step after which extracted information is used for finding optimal assignments. The latter algorithms are to the most extent based on deformable models [102, 111]. The general concept is to use models that have been extracted from frame t as input to the detection procedure for frame $t + 1$ while keeping track of the explicit connection between models. Such approaches are particularly useful for image data from experiments with an immobile specimen or a high imaging frequency compared to the mobility. A comprehensive benchmark of theoretical approaches as well as implementations has been conducted in context of a Particle Tracking Challenge [112] at the International Symposium on Biomedical Imaging (ISBI) 2012. The benchmark for the challenge and the results have previously been published by Maška *et al.* [113] and Chenouard *et al.* [114], respectively.

Due to the challenging image data from MSCC, we have developed a highly opti-

mized cell detection pipeline for SCA of *C. glutamicum*. An integration of the previously presented cell detection approach (cf. Chapter 4) is only possible with approaches that have decoupled the cell detection and the tracking step. Within the category of tracking after detection, approaches can be further structured according to the mathematical framework they are based on: linear program/assignment problem formulations, probability-based Bayesian approaches, straight forward nearest-neighbor linking. The latter algorithms are underperforming due to limited amount of information considered for finding assignments [114]. The former methods are using multiframe and/or multitrack information, effectively incorporating additional data and, hence, improving the tracking results.

In the following section, I will present two approaches for finding optimal assignments that are based on an assignment problem and a Bayesian formulation, respectively.

5.1 Single-Particle Tracking

One method for solving the assignment problem has been proposed by Jaqaman *et al.* [115]. The proposed procedure is composed of two consecutive steps, namely the frame-to-frame linking and the global linking of tracks. While the approach is formulated in a general manner for the tracking of particles and is referred to as single-particle tracking (SPT) rather than cell tracking, the ability to handle merging as well as splitting events makes it applicable for investigations of cellular development.

5.1.1 Frame-to-Frame Linking

In the first step of the algorithm, assignments are established between objects (so-called particles) in two consecutive frames, discarding information from other frames. The resulting assignments are termed track segments, implying that they have not been linked together to complete tracks. A completed track contains the development of an individual cell over the course of an experiment.

In order to find potential correspondences and create track segments, three types of assignment are considered:

1. A cell in frame t is assigned to a cell in frame $t + 1$ (“one-to-one”/“continue”)
2. A cell in frame t does not have a correspondence in frame $t + 1$ (“track end”)
3. A cell in frame $t + 1$ cannot be linked to a cell in frame t (“track start”)

The computation of optimal assignments is based on the similarity of particles in two consecutive frames. The more similar two particles are, the lower is the cost for an

assignment. In the most basic case, the cost is a function of the spatial distance between the centroids of two cells c_i and c_j:

$$l_{ij} = \left\| c_i - c_j \right\|^2 . \tag{5.1}$$

From the calculated costs for each possible assignment between frames t and $t+1$ a cost matrix C_{local} is created (structure shown in Figure 5.2). In addition to linking two cells (upper, left block of the matrix) the aforementioned three types of correspondences (i.e., one-to-one assignments, segment starts, and segment ends) are also taken into account. The costs for starting or ending a segment are determined using information about the costs of one-to-one assignments.

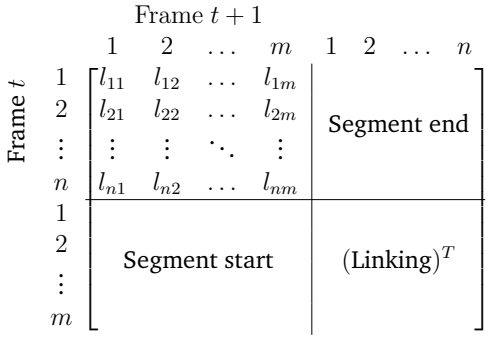


Figure 5.2: Cost matrix C_{local} for linking cells in two consecutive frames. The matrix contains four blocks. The upper left block contains the cost l_{ij} for linking cells of frame t and $t+1$. The *Segment end* and *Segment start* blocks are alternatives, if no direct correspondence is found. The lower right *filling block* is the transposed linking block, to fulfill the requirements for an LAP formulation (adapted from [115]).

Entries are considered not feasible according to a defined cut-off value. Usually this cut-off is defined by a maximum distance that cells can move between two subsequent frames, such that an assignment of cells which are unrealistically far apart is not taken into account. In practice, this helps to reduce the memory consumption in the implementation of the LAP.

Having established the cost matrix C_{local}, shown in Figure 5.2, it is the goal to find the set of most fitting assignments. This is achieved by formulating an objective function with side constraints in a LAP framework. In this context, the combination of assignments with the minimal sum of costs is searched for by Equation 5.2. The following formulation states the mentioned goal as a general LAP:

$$\hat{A} = \underset{A_{ij}}{\operatorname{argmin}} \sum_{i,j} A_{ij} C_{local,ij} \tag{5.2}$$

subject to

$$\sum_i A_{ij} = 1, \forall j \qquad \text{and} \qquad \sum_j A_{ij} = 1, \forall i. \tag{5.3}$$

In Equation 5.2, the assignment matrix A is defined as

$$A_{ij} = \begin{cases} 1 & \text{if assignment is selected} \\ 0 & \text{else} \end{cases} \tag{5.4}$$

The feasibility constraints established by Equation 5.3 exclude the case that more than one possible assignment is selected for a cell in frame t and $t+1$, respectively. Equation 5.2 and Equation 5.3 poses a constrained linear program (LP). This optimization problem is solved with well established techniques and implementations, for example the Simplex algorithm [116, pp. 864-879]. The formulation of this linking step as a LAP, however, allows the use of optimized algorithms, i.e., the Munkres-Kuhn algorithm [117] or the improved Jonker-Volgenant algorithm [118]. The latter has improved the runtime of the original Hungarian algorithm from $\mathcal{O}(n^4)$ to $\mathcal{O}(n^2 \log n)$ by leveraging the sparseness of LAPs.

5.1.2 Global Track Linking

The frame-to-frame linking step establishes correspondences between pairs of consecutive frames of a complete image sequence. Subsequent to this step, the role of the global track segment linking is to join track segments where necessary, but also to split or merge tracks. This is necessary to handle situations where no correspondence is found but a track segment start might be connected with a track segment for a split. Possible explanations for this are objects leaving the image or moving to far, resulting in a cut-off. Thus, the tracks (several consecutive correspondences in a row) that have been computed during the previous step have to be linked to each other, taking into account the information of the whole image sequence. Additionally, splitting and merging (which is not biologically interpretable) events have to be determined.

Therefore, global optimization is performed in the temporal domain. This is achieved by formulating the problem as an LP. In contrast to C_{global}, the cost matrix C_{global} (see Figure 5.4) in the global track segment linking step considers the following hypotheses (visualized in Figure 5.3):

Close gap: Connect one track segment end with a track segment start where the cells do not necessarily have to be in subsequent frames

Merge track segments: Connect a track segment end to a cell in another track segment

Split track segments: Connect a track segment start with a cell from another track segment

Initiate or terminate tracks: Keep a track segment start or track segment end, respectively

Figure 5.3: Hypotheses considered in the global track linking step of the SPT approach.

The cost for closing a gap is computed from the distance between cell J at the end of track segment J and the cell K at the start of track segment K:

$$g_{JK} = \|c_J - c_K\|^2 . \tag{5.5}$$

The costs for merging and splitting are also based on information about the distance between the proposed merging and splitting points, respectively. Per convention, J denotes the start or end of a track segment J and K denotes a middle point of track K:

$$m_{J,K} = \|c_J - c_K\|^2 = s_{J,K} \tag{5.6}$$

Hence, the merging cost $m_{3,10}$ in C_{global} (cf. Figure 5.4) denotes the merging of the track segment end 3 and track segment 10 at cell 20 of this segment. The same notation is used for splitting in C_{global}.

Initiation and termination cost of tracks are computed according to alternatives, leaving the track segments unchanged. For details on the computation, the reader is referred to [115].

Hence, the global cost matrix C_{global} contains blocks for the aforementioned possibilities. The general structure of C_{global} is setup to adhere to the LAP framework. Thus, the top row as well as the left column contain indexes for all available track segments that have been created during the frame-to-frame linking step (cf. Subsection 5.1.1). This also applies for the bottom row and the right column of the cost matrix.

The middle row and middle column have indexes for all cells that are not located a track segment start or track segment end in conjunction with a track segment identifier. In essence, the index 1/5 denotes the cell c_5 in track 1.

The general concept of the matrix construction is similar to the construction of C_{local}. For each row (or column) one assignment has to be selected. Therefore, each track segment of the top block is assigned a gap closing, a merging event with a cell from another track segment, or it can be terminated. The termination is achieved by assigning the track segment to itself in the top right block of C_{global}.

C_{global} is constructed such that the problem formulation is equivalent to Equation 5.2 with Equation 5.3. In this step, however, all track segments throughout the entire image sequence compete which each other, thus, the solution to the LAP is global in time.

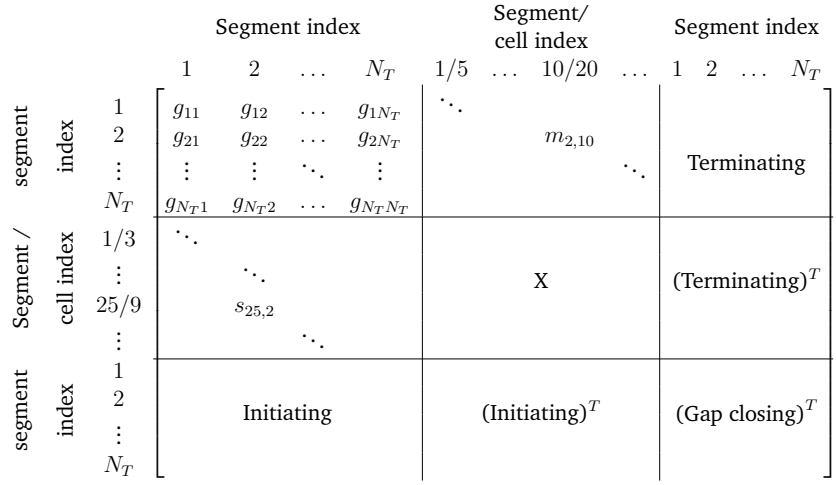


Figure 5.4: Cost matrix C_{global} for global track linking (adapted from [115]). $g_{i,j}$ denotes the gap closing costs (cf. Equation 5.5), $m_{J,K}$ the merging cost (cf. Equation 5.6), and $s_{J,K}$ the splitting cost (cf. Equation 5.6). The two blocks for terminating and initiating tracks compete with the other alternatives. The remaining blocks are constructed such that the matrix can be used in the LAP framework.

5.2 Probability-Based Approach

The approach proposed by Al-Kofahi *et al.* [119] is based on the computation of likelihood values for an ensemble of possible assignments between cells in frame t and $t+1$. The likelihoods are formulated as multivariate Gaussian distributions, based on a comparison between the cells for which a correspondence is to be established.

To be able to evaluate the similarity or dissimilarity of cells, a characterization is needed. For each detected cell c_i^t, some easy to calculate measures are computed in each image t. This includes the cell's center of mass $(x, y)_i^t$, the cell area a_i^t, the eccentricity e_i^t, major axis length l_i^t, and orientation θ_i^t (defined as angle between the major axis and the x-axis). The visual representation of the cell descriptors is shown in Figure 5.5. The characterization and, hence, the tracking is based on the so-called feature vector $f_i^t = ((x, y)_i^t, a_i^t, e_i^t, l_i^t, \theta_i^t)$.

Different hypotheses are tested and, therefore, different likelihoods have to be computed. A cell can either

1. *move and deform* (Equation 5.7), or
2. *divide* into two new cells (Equation 5.8).

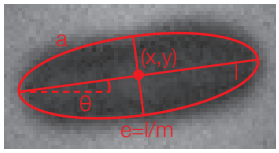


Figure 5.5: Cell descriptors from which the feature vector f_i^t of a cell c_i^t is constructed.

These likelihoods are modeled as multivariate Gaussian distributions. Hence, the probability for the move hypothesis is formulated:

$$p_{move}(f_i^t, f_j^{t+1}) = \frac{1}{\sqrt{(2\pi)^N |\Sigma|}} \exp\left\{-\frac{1}{2}(d_{ij}^t - \mu)^T \Sigma^{-1} (d_{ij}^t - \mu)\right\} \tag{5.7}$$

where N denotes the number of elements in the feature vector and d_{ij}^t is the absolute difference between the two feature vectors $d_{ij}^t = |f_i^t - f_j^{t+1}|$ (i.e., the vector of absolute differences between corresponding cell descriptors). μ denotes the mean value and Σ the covariance matrix of the difference vector d_{ij}^t. Both parameters have to be calculated for a training set, but can usually be assumed constant. Hence, the estimation is done for one type of experiment and can then be used for experiments where the imaging setup and cell behaviors are similar to the training sequence.

The formulation of the likelihood for the second hypothesis, namely that one cell, characterized by the feature vector f_i^t, divides into two new cells with feature vectors f_j^{t+1} and f_k^{t+1}, is similar to Equation 5.7:

$$p_{divide}(f_i^t, f_{jk}^{t+1}) = \frac{1}{\sqrt{(2\pi)^N |\Sigma|}} \exp\left\{-\frac{1}{2}(d_{ijk}^t - \mu)^T \Sigma^{-1} (d_{ijk}^t - \mu)\right\}. \tag{5.8}$$

The computation of the difference vector d_{ijk}^t, however, has to be adapted: an artificial cell c_{jk}^{t+1} is created which comprises pixel of both cells c_j^{t+1} and c_k^{t+1}. Consequently, a feature vector f_{jk}^{t+1} is computed for this dummy cell as well as the difference vector that is computed as $d_{ijk}^t = |f_i^t - f_{jk}^{t+1}|$.

The problem of finding the most likely hypothesis is stated as a LP. For this reason a matrix D is constructed, such that each possible hypothesis for two frames generates one row in the matrix (see artificial example in Table 5.1). The first columns of the matrix denote the cells of frame t, followed by the cells in frame $t+1$. All entries in one row are zero except for the columns of the cells that are part of the hypothesis. For this reason, the likelihoods computed in the previous step are split equally between two cells, if the hypothesis is *move and deform*. In case of a *division* event, the initial cell is given half the likelihood and each of the others is assigned $25\,\%$.

In a LP framework, the goal is stated as finding a selection vector $\hat{\mathbf{x}}$ that maximizes the overall sum of likelihoods, but only selects one hypothesis per cell. The mathematical

	Frame t					**Frame $t+1$**					
	1	**2**	**3**	**4**	**5**	**a**	**b**	**c**	**d**	**e**	**f**
$1 \rightarrow$ **a**	0.43	0	0	0	0	0.43	0	0	0	0	0
$1 \rightarrow$ **b**	0.09	0	0	0	0	0	0.09	0	0	0	0
$1 \rightarrow$ **c**	0.08	0	0	0	0	0	0	0.08	0	0	0
$1 \rightarrow$ **f**	0.08	0	0	0	0	0	0	0	0	0	0.08
$2 \rightarrow$ **b**	0	0.18	0	0	0	0	0.18	0	0	0	0
$2 \rightarrow$ **f**	0	0.18	0	0	0	0	0	0	0	0	0.18
$2 \rightarrow$ **c**	0	0.11	0	0	0	0	0	0.11	0	0	0
$3 \rightarrow$ **c**	0	0	0.42	0	0	0	0	0.42	0	0	0
$3 \rightarrow$ **f**	0	0	0.10	0	0	0	0	0	0	0	0.10
$4 \rightarrow$ **d**	0	0	0	0.46	0	0	0	0	0.46	0	0
$5 \rightarrow$ **e**	0	0	0	0	0.44	0	0	0	0	0.44	0
$1 \rightarrow$ **a, b**	0.02	0	0	0	0	0.02	0.02	0	0	0	0
$2 \rightarrow$ **b, f**	0	0.28	0	0	0	0	0.28	0	0	0	0.28
$3 \rightarrow$ **c, f**	0	0	0.04	0	0	0	0	0.04	0	0	0.04

Table 5.1: Excerpt of the matrix referred to as D in Equation 5.9. Rows correspond to the different hypotheses stated in the first column of the matrix. Details on constructing the matrix are described in the text (adapted from [119]).

formulation of the objective function and the constraints are formulated as:

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmax}} (D\mathbf{v})^T \mathbf{x} \tag{5.9}$$

subject to

$$B^T \mathbf{x} \leq \mathbf{v}$$

where $\mathbf{v}$ is a vector with the number of possible hypotheses in length and all entries 1. Furthermore, the companion matrix B is constructed from D by keeping an entry if it is zero in D and setting it to 1 otherwise. With that restriction, only one hypothesis is selected for each row and thereby each cell.

An example for an assignment matrix D in combination with calculated likelihoods of hypotheses is given in Table 5.1. The likelihoods have been adapted from the original publication [119] for illustration purposes.

5.3 Conclusions

Although the challenges for cell tracking approaches are manifold in microfluidic LOC devices, the aforementioned approaches are promising concepts. These approaches have been evaluated with respect to their applicability to the acquired images at IBG-1 as well as the possibility of integration into the existing ecosystem of the already established modules of the image analysis pipeline.

The probability-based approach offers flexibility with respect to used cellular characteristics and makes the integration of additional cellular characteristics straight-forward. However, integration of prior information is hard do achieve in this framework. In addition, an implementation of the proposed algorithm was not available for further benchmarking.

In contrast to the approach proposed by Al-Kofahi *et al.*, the SPT algorithm was available as an original implementation *u-track* [120] for preliminary tests. The approach was furthermore implemented, adapted, and optimized by Jean-Yves Tinevez as a Fiji (cf. Section 6.1) plugin. TrackMate [121] is actively developed and maintained by the Fiji community. Furthermore, it integrates well with the established ecosystem at IBG-1.

6 Implementation

The previously presented image analysis pipeline has been implemented as a suite of plugins and macros for the image processing and analysis software ImageJ/Fiji. The following sections will introduce ImageJ/Fiji, provide a general overview of the code base and highlight selected implementation details.

6.1 ImageJ

ImageJ is an open source framework for image processing and image analysis tasks. While it started out in 1997, more as a tool than a framework, it has developed into providing the (bio)image analysis community with a scaffold for a variety of tasks. The development approach is community-driven (i.e., academic as well as industrial contributors) and, hence, many contributors have implemented a plethora of plugins (>100) for additional functionality. Furthermore, the main contribution of ImageJ is that it provides a user interface and the means to load and save images. Additionally, many basic image processing tasks are already implemented and can be accessed conveniently through the (barely changing and thus consistent) user interface and via a custom macro language.

In recent years, the development of ImageJ has moved into a new direction that was later termed ImageJ2. This development was mainly driven by the developers of Fiji [122] who had previously set out to bundle create a distribution of ImageJ with useful plugins bundled and configured. Parallel to the advent of imglib2 [123], the underlying data model of ImageJ was greatly improved with generic data structures that were not restricted by implementation details of the legacy/vanilla ImageJ as developed by Wayne Rasband.

While ImageJ2 features newer user interfaces, they are not loaded by default. Hence, the changes are mostly hidden from the end user but open up new opportunities for developers. The underlying plugin framework underwent a major rewrite (starting in 2012, stable in 2014) in which much of the functionality was refactored to SciJava [124]. The plugin mechanism is now based on services that can provide functionality and are

discovered automatically at runtime. This is an important and major step forward in that plugins can now provide functionality for other plugins and this functionality can be distributed in *Plugin A* but is discovered at runtime by *Plugin B*. We are leveraging this functionality to extend other plugins in the ImageJ ecosystem by bundling and distributing one jar-file, which eases the use for end users. Furthermore, we do not have to fork other plugins for custom implementations, if they have been designed and implemented to discover new functionality using ImageJ's service architecture.

ImageJ is available in three different flavors:

- ImageJ, also ImageJ1
- ImageJ2
- Fiji

ImageJ1 is the spiritual successor of NIH Image (development started in 1987 [125]).

Although ImageJ2 is a major rewrite and advancement of ImageJ, a compatibility layer has been implemented. This layer enables the execution of ImageJ1 plugins in ImageJ2 context as well as on-the-fly conversion of ImageJ2 datasets for use with ImageJ1 plugins. Furthermore, core functionality that is not necessarily only usable in the image processing context has been modularized for use in other projects.

Fiji and ImageJ2 use the Bioformats library [126] for both loading and writing all major image formats used in the (bio)imaging community.

6.2 General Overview

Figure 6.1 shows the general structure of the code base of the proposed image analysis pipeline. In the remainder of this thesis, the implementation of the image analysis pipeline will be referred to as Jülich Next Generation Lineage Extractor (JuNGLE).

MasterPlugin When an image is open, the `MasterPlugin` shows the user interface in which the steps of the image analysis pipeline can be selected for execution (cf. Figure 6.2). Depending on the activated modules, parameters are set by the user. This includes the profile that is used for the execution of the `SegmentationPlugin` (i.e., in Figure 6.2 "Default") as well as filter settings for the splitting of dense populations, as described in Section 4.6.

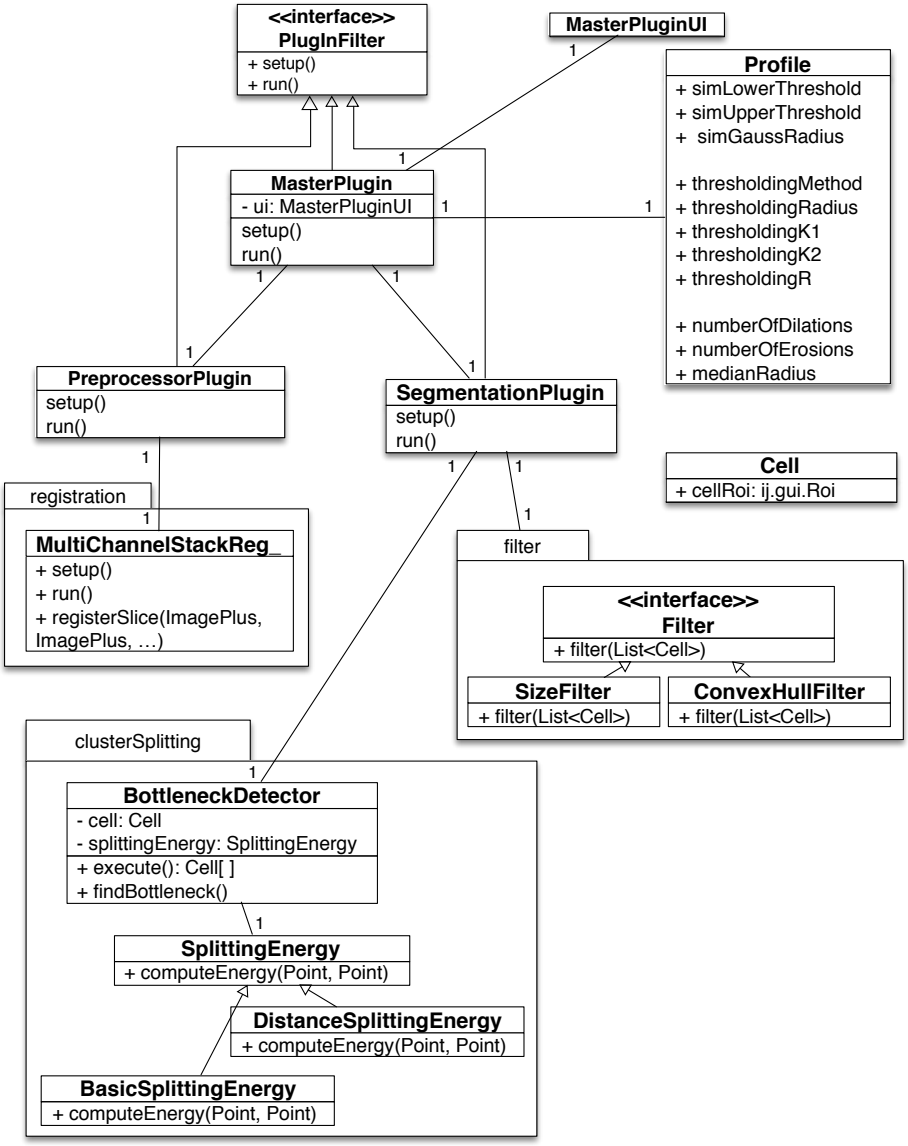


Figure 6.1: Main components of the implemented image analysis pipeline as ImageJ plugin in a UML diagram.

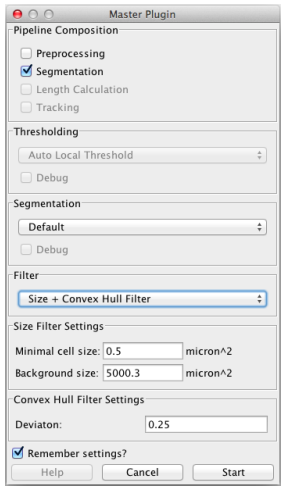


Figure 6.2: Central user interface component for controlling parameters for the preprocessing procedures as well as the cell detection step. In addition to selecting which parts of the pipeline are executed, segmentation profiles (Section 6.5) as well as filter (Section 4.5) can be configured.

PreprocessorPlugin The `PreprocessorPlugin` implements the workflow for processing raw image data according to Chapter 3. The `run()` method creates and executes the `MultiChannelStackReg_` plugin for registering images. Afterwards, the user has either manually selected a ROI, in which case the image sequence will be cropped and rotated using this ROI, or the box detection procedure is activated. Once the growth site has been detected, it is, just like in the manual case, cropped and rotated.

The box detection procedure consists of three steps. A custom edge detection is implemented in `EdgeDetector` and is described in more detail in Section 6.4.

The inner while loops (the classification) of Algorithm 2 are implemented in the `*BoxClassifier` classes. `RatioBoxClassifier` implements an improved classification scheme, that is not based on the absolute size of growth sites, but rather on the ratio of length to width. While this approach is more compute intense, it will also detect boxes in zoomed out images where the absolute dimensions might not be available for `DimensionBoxClassifier`.

SegmentationPlugin The `SegmentationPlugin` applies the in Chapter 4 presented cell detection pipeline to all images of an image stack. Since the segmentation of one image is independent of the rest of the sequence, processing is executed in parallel threads. Due to the nature of this parallelization, the memory consumption of

`SegmentationPlugin` scales linearly with the number of threads that is used for processing.

Due to the branching in the cell detection pipeline (cf. Figure 4.4), each thread uses at least twice the memory of the input image. Furthermore, images are upscaled before processing to improve the segmentation results. Hence, each thread minimally uses eight times the memory of the input image. During the processing, temporary images of the same size as the upscaled input image have to be kept in memory.

6.3 MultiChannelStackReg_

The original StackReg plugin [127] by Philippe Thévenaz that implements the pyramid approach for image registration (cf. Subsection 3.1.2) only works for single-channel image stacks. The main functionality of the original implementation was to register a single stack using the TurboReg plugin [128], using the first image of a stack as template. During this project, the necessity for multi-channel registration as lead to an extension of its functionality to work with multi-fluorescence stacks.

The computation of the best transformation is achieved in the first channel of an image only. The transformation for each frame is subsequently applied to the other channels of the image sequence with a recomputation of the transformation.

6.4 EdgeDetector

The `EdgeDetector` convolves the image with a kernel created from a phase contrast image. This kernel has been generated by manual selection of an edge image of a growth site under typical experimental conditions. Hence, regions where this kernel is found in the image will produce a high value in the resulting edge image. This edge image is globally thresholded such that regions with a similarity score higher than 0.95 are considered edges. This procedure can easily be replaced with a standard method like Canny's method for edge detection. It has, however, proven to be useful to take into account how the CAD designed growth sites appear in the phase contrast image as described in Section 3.2.

6.5 Processing Profiles

All modules of the pipeline have been implemented with flexibility in mind. This flexibility should not only be benefit developers, working with the code, but more for experimentalists to refine their results. Hence, the pipeline allows for the creation of so-called profiles that store parameters. There are many prebuilt profiles with parameter sets that have been optimized and tested for *C. glutamicum* with a specific experimental setup.

The user can select profiles before starting the execution of JuNGLE. This is shown in Figure 6.2, where “Balaban” is selected. This is a profile that has been optimized for a specific sets of experiments, with a high temporal resolution and low exposure.

In addition to those profiles, the settings that are used for the execution, are stored locally for the processing of further images with the same settings.

6.6 Splitting Tool

The splitting tool is one of several macros that are part of the JuNGLE suite. Once the tool is selected, it is activated for the currently active image and can be used to manually curate segmentation results (cf. Figure 6.3). Because the segmentation results are stored in an overlay/regions of interest, it is also possible to use the all available functionality of ImageJ for this. The splitting tool, however, provides a convenient way to select cells and split them along a line with some visual help with minimal user interaction.

In addition to splitting, it is also possible to merge two cells with the help of the tool. To that end, the user has to select multiple cells while pressing *Shift*. When the *Shift* key is released, the segments are merged together to form a new cell.

Furthermore, the cells follow a specific naming convention. Therefore, users are not advised to add manual selections to the overlay, but use a macro that generates unique names for ROIs. Again, all available tools can be used for the selection, when “A” is pressed, this ROI is added to the segmentation results.

6.7 Snakes

The models of ellipse- and rod-reproducing snakes have been implemented as a standalone plugin for ImageJ/Fiji. We have decided to pack it as standalone plugin, so that it can be used independently of the image analysis pipeline. When executed, the plugin uses a polygon selection and the active image as input and requires parameters through an user interface (cf. Figure 6.4). Once the mandatory parameters are set, the iterative procedure starts and energies are logged for the user to observe the progress of the computation.

In addition to execution of the plugin the standalone way, it has also been integrated with JuNGLE. The optimization of a ROI is started with a keyboard shortcut in an output image of the cell detection module. This is possible, because parameters for the snakes can also be defined through an API that allows the plugin to be executed in headless mode.

More details on the implementation are provided in [98].

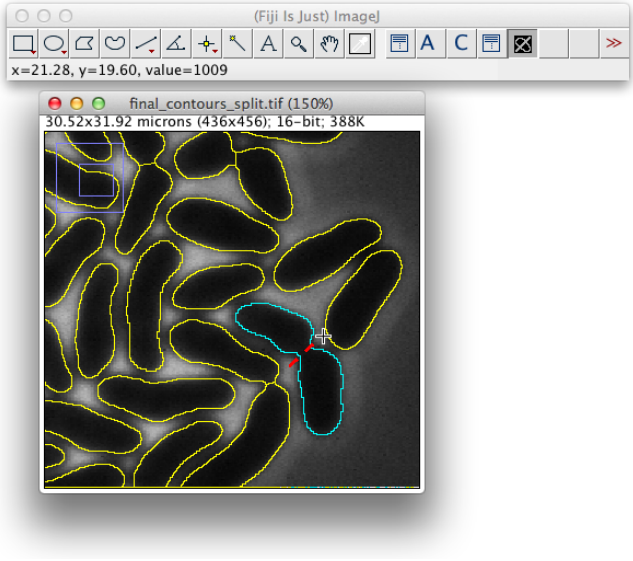


Figure 6.3: Screenshot of Fiji with the activated JuNGLE Splitting Tool. The population has been processed, but contains an error. The Splitting Tool shows the red, dashed line along which the cell will be split. The blue contour denotes the currently selected cell that is about to be split.

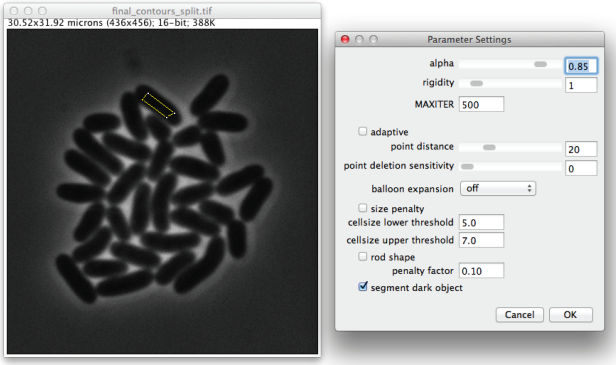


Figure 6.4: Screenshot of an image with an active polygon ROI. Snakes Plugin is started and requires parameter input.

6.8 Link to TrackMate

TrackMate is a full-fledged tracking framework that is based on the Single Particle Tracking approach by [115] (cf. Section 5.1). Improvements to the proposed implementation have been made to speed up computations and reduce memory consumption during the computations.

Internally, TrackMate establishes a data model that consists of `Spots` and `Tracks`, where the connection between the spots is stored in tracks. The fundamental spot descriptors, i.e. location, are readily available with the package. There is, however, the possibility to enhance TrackMate's functionality by adding new or extending existing modules of the pipeline.

Because TrackMate has been built with extensibility in mind, it is possible to add new `SpotDetectors` (extract `Spots` from image sequences), `Views` (visualize `Spots` and `Tracks` on the input image), as well as `Trackers` by implementing provided interfaces. In addition to the implementing classes, `Factory` classes provide additional information about the implementations, i.e., which features are provided. These factory classes are annotated such that they can be discovered by TrackMate (through SciJava, cf. Section 6.1) at runtime.

6.8.1 OverlayDetector

As already mentioned, TrackMate provides `Detectors` that take an image sequence as input and extract a collection of `Spots` from the image sequence. I have implemented a custom detector that converts ImageJ regions of interest to `Spot` object, effectively linking those together.

6.8.2 SpotAnalyzer

Since a connection has been established between `Spots` and ROIs, it is possible to use built-in functionality of ImageJ to extract information about the ROI and store it in the according `Spot` object. To that end, a `SpotMeasurementAnalyzer` has been implemented that extracts cell area, cell length, and centroids for each spot. In addition to the measurements that are extracted from the bright-field image, a `SpotFluorescenceAnalyzer` is available, that extracts the fluorescence intensity values (absolute, mean, standard deviation) for available fluorescence channels.

6.8.3 Actions

Several so-called `Actions` have been implemented that can be executed directly after the tracking has been finished. Most actions have been developed in the context of the applications that are going to be presented in Part III. Since, more complex analysis

tasks with time-resolved data are supposed to be executed with *Vizardous*, the actions are rather small scripts that can, for instance, be used for screening purposes.

The `FluorescenceThresholdScreener` has been developed to test if a cultivation contains cells that exceed specific fluorescence intensity thresholds. For this action, it is not necessary to have executed the tracking step already. It can rather be used for checking if this cultivation contains an occurrence of a rare event and decided based on the output if further processing is advisable. The `FluorescenceMaximumScreener` extracts the maximum fluorescence intensities that are encountered in the analyzed cultivation.

In contrast to the previously described screening actions, the most import action is the `JungleExporter`, which exports the tracks and extracted cellular characteristics to the PhyloXML and MetaXML format for visualization and analysis with *Vizardous* (cf. Chapter 8).

7 Quality Assessment

We assess the quality of the proposed methods and implemented modules with respect to three different factors: *a*) correct detection of cells, *b*) influence of contrast changes on cell detection results, and *c*) correct assignments of cells in two subsequent images. These factors enable the assessment of the quality of the proposed cell detection procedure as well as the tracking approach and highlight challenges.

To that end, the number of false positive detections of cells with respect to a manually curated gold standard was computed. Gold standards were generated during the evaluation of the respective datasets for the applications described later in this thesis (cf. Part III). The cell detection procedure was executed using predefined profiles (cf. Section 6.5) optimized for the respective dataset (i.e., mainly the resolution of image). We have selected the ratio of correctly detected cells to the overall number of cells (that has been determined manually) as a criterion for assessing the quality of the cell detection module. In the same spirit, the ratio of correct assignments between cells to the overall number of assignments has been determined. The assessment of the quality of the tracking module is based on manually curated lineage trees for *C. glutamicum* datasets.

With respect to the applications (Part III), it is possible to categorize the data according to the spatial as well as the temporal resolution (cf. Figure 7.1). Furthermore, the exposure time heavily influences the SNR as well as the gray value difference between background and foreground pixels.

All below mentioned computations have been executed on an Intel©Core™i7-2600 (3.4 GHz $\times$ 4) central processing unit (CPU) and 16 GB of random access memory (RAM). The benchmarks are based on the implemented plugin of the modules (cf. Chapter 6) executed on Ubuntu 12.04.

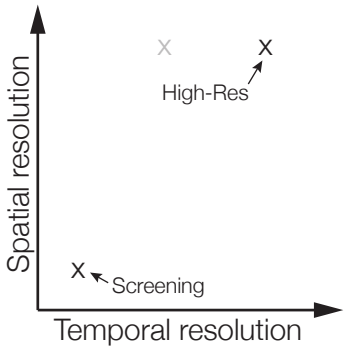


Figure 7.1: Schematic of the location of the used datasets in the temporal/spatial resolution plane. One dataset has a low temporal and low spatial resolution. A second one offers a high spatial as well as temporal resolution.

7.1 High Temporal Resolution

We have generated a set of image sequences with a high image acquisition frequency ($\Delta t = 2\,\mathrm{min}$) to accurately determine division times of *C. glutamicum* ATCC 13032 on complex brain-heart infusion (BHI) medium. This dataset is called *HighRes* in the following. It has been used for general growth studies and a systematic comparison of quantification procedures for microbial growth in Chapter 10. Figure 7.2 shows two images from different growth sites of the dataset.

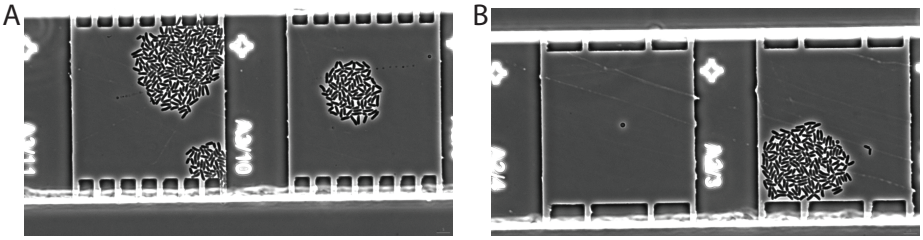


Figure 7.2: Exemplary raw images from the *HighRes* dataset. Each image has two growth sites in the field of view. **A.** Position 16, $4.8\,\mathrm{h}$ into the experiment. **B.** Position 33 at $5.8\,\mathrm{h}$.

7.1.1 Preprocessing

Spatial shifts between two frames are in the order of $1\,\mu\mathrm{m}$ to $3\,\mu\mathrm{m}$ for this dataset. Hence, the registration of image stacks works without problems. The computational demand, however, is also increased in comparison with sequences of lower temporal resolution because of the increased number of frames that have to be processed.

In this special case, the field of view does not only contain one, but two MGCs. Hence, we had to split the images before applying the box detection module. Once images are separated into individual MGCs, manual selection of growth sites had to be done for 3 out of 41 growth sites for this dataset (not all of the 41 available growth sites have been used in this or later studies).

The runtime of the preprocessing procedure depends on the ratio of moving and immobile contents in an image. Registration takes about $2\,\mathrm{s}$ per frame in the beginning of a sequence. When the number of cells in a growth site increases, registration times increase to $5\,\mathrm{s}$ per frame when an MGC is filled with more than 300 cells. A complete image sequence is preprocessed in $10\,\mathrm{min}$ to $15\,\mathrm{min}$ depending on the length of the sequence and the size of the images.

7.1.2 Cell Detection

Due to the high spatial resolution of the images ($>15\,\mathrm{px}/\mu\mathrm{m}$) and a reasonable signal-to-noise ratio, the cell detection module performs well. We have computed statistics of how many cells had to be manually corrected during the generation of a gold standard. For the evaluated growth sites the percentage of cells that were detected (and split) correctly is around $98\,\%$. The complete set of statistics is provided in Table 7.2. Exemplary images extracted from the output of the cell detection module for growth site 16/2 are shown in Figure 7.3.

Table 7.1: Quality of the cell detection module for different cultivations from the *HighRes* dataset. The number of assignments denotes the number of all available assignments between cells. Corrected assignments have been manually curated to be correct.

DatasetID	Number of Cells	Corrected Cells	% Correct
03/1	12279	216	0.9824
16/2	15743	102	0.9935
33/2	8003	99	0.9876
35/1	7412	51	0.9931
39/2	5344	123	0.9770

Figure 7.3C shows a *C. glutamicum* colony of high cell density. When zooming on a later frame in the image sequence (cf. Figure 7.4), an underestimation of the cell area is detectable near the center of the population. This is on the one hand due to the set of parameters that is not able to capture image characteristics for the complete image. On the other hand, the image contrast changes harshly near the population center. A possible explanation for this observed effect is the applied contrast enhancement technique for bright field images (phase-contrast).

The overall runtime for one image sequence ranges from $20\,\mathrm{min}$ to $40\,\mathrm{min}$. The most influential factor for the runtime of the cell detection module is the cluster split-

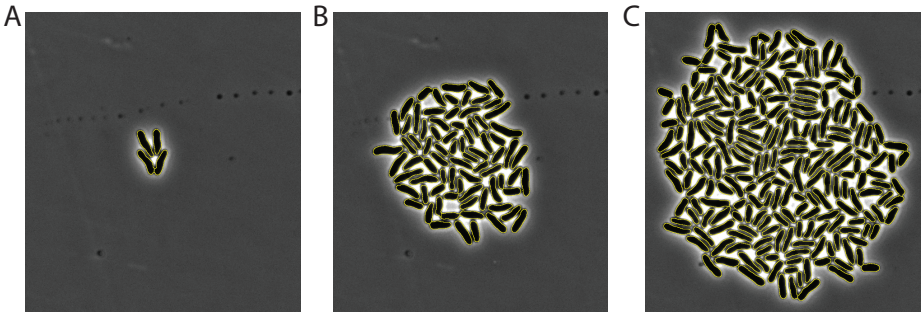


Figure 7.3: Results of the cell detection module for dataset with ID 16/2. Time points of the images are $1\,\mathrm{h}$ (**A**), $4.6\,\mathrm{h}$ (**B**), and $5.9\,\mathrm{h}$ (**C**).

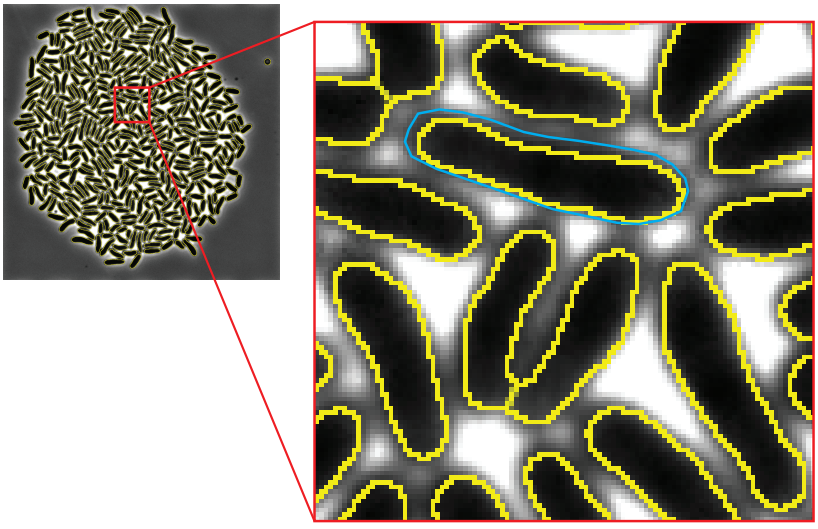


Figure 7.4: The overview image on the left shows frame 200 from the image sequence of growth site 16/2. The panel on the right shows a zoom to the center of the population. Yellow contours denote the results from the cell detection module, while the blue contour shows a manually set cell contour for reference.

ting approach. The better the presented binarization procedure performs, the fewer optimizations have to be executed (cf. Section 4.6).

7.1.3 Tracking

With an increase in cell density in a population the frequency of misassignments increases (cf. Table 7.2). Especially near the end of a cultivation, the distance traveled between two frames increases for cells that are located near the perimeter of a population. The volume increase of cells in the center forces the former cells to the outside. In the worst case (heavily depending on the image acquisition frequency) the traveled distance is a multiple of an average cell length.

Table 7.2: Quality of the tracking approach for different cultivations from a dataset with a high temporal resolution. The number of assignments denotes the number of all available assignments between cells. Corrected assignments have been manually curated to be correct.

DatasetID	Number of Assignments	Corrected Assignments	% Correct
03/1	12320	167	0.9864
16/2	15732	194	0.9877
33/2	7996	87	0.9891
35/1	7239	59	0.9918
39/2	5342	45	0.9916

The runtime of the tracking procedure for this dataset is in the sub-second range for each growth site.

7.2 Low Temporal and Spatial Resolution

We have generated a dataset that is located on the opposite side of the spectrum in Figure 7.1 for a screening application (referred to as *Screening* dataset). The low spatial resolution of images ($\Delta t = 15\,\mathrm{min}$) is due to the fact, that the data has been generated using a subpar CCD camera. Furthermore, to test as many medium conditions as possible in parallel (cf. Chapter 11), the image acquisition frequency has to be lowered to increase throughput.

Nevertheless, the influence of lower resolutions on the preprocessing module is marginal. Information from Subsection 7.1.1 apply as well for this low resolution dataset. Runtimes are, however, comparably shorter because of the lower spatial resolution of images.

7.2.1 Cell Detection

Statistics of how many cells had to be manually corrected during the generation of a gold standard have been prepared for the *Screening* dataset. The percentage of cells

that were detected (and split) correctly ranges from $75\,\%$ to $98\,\%$. Unsteady image quality between different cultivation conditions determines the outcome of the cell detection module. The complete set of statistics is provided in Table 7.4. Exemplary images extracted from the output of the cell detection module for different cultivation conditions are shown in Figure 7.5.

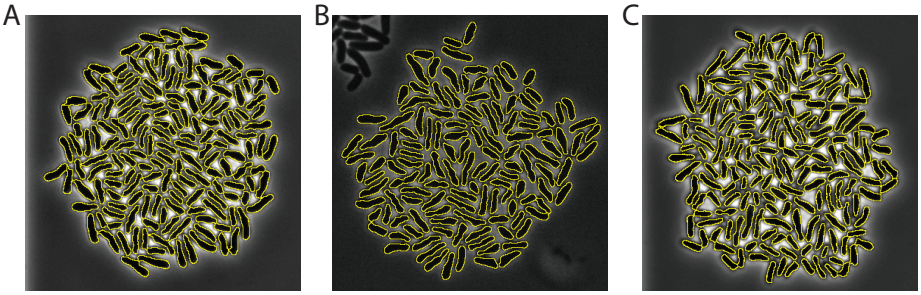


Figure 7.5: Results of the cell detection module for datasets with IDs nd055/100 (**A**), nd013/15 (**B**), and nd070/20 (**C**). Images have been extracted from sequences at $5\,\mathrm{h}$ (**A**), $4\,\mathrm{h}$ (**B**), and $5\,\mathrm{h}$ (**C**).

Table 7.3: Quality of the cell detection module for different cultivations from the *Screening* dataset.

DatasetID	Number of Cells	Corrected Cells	% Correct
nd055/100	1428	293	0.7948
nd055/102	2247	58	0.9742
nd055/105	1853	114	0.9385
nd013/2	684	172	0.7485
nd013/14	656	107	0.8370
nd013/15	577	103	0.8215
nd070/10	658	43	0.9347
nd070/12	886	71	0.9199
nd070/20	977	146	0.8506

7.2.2 Tracking

The percentage of correctly assigned cells ranges between $70\,\%$ and $93\,\%$ (Table 7.4). One possible explanation for the broad range, in the particular case of the *Screening* dataset, is the difference in growth rates of the populations based on medium composition. Hence, the constant imaging interval of $\Delta t = 15\,\mathrm{min}$ will produce less samples of generations of fast growing cells than of slower growing cells. Thus, the already mentioned movement between images is influenced by the growth characteristics as well.

Table 7.4: Quality of the tracking approach for different cultivations from the *Screening* dataset. The number of assignments denotes the number of all available assignments between cells. Corrected assignments had to be manually corrected during data curation.

DatasetID	Number of Assignments	Corrected Assignments	% Correct
nd055/100	1417	103	0.9273
nd055/102	2234	279	0.8751
nd055/105	1828	218	0.8807
nd013/2	653	190	0.7090
nd013/14	636	174	0.7264
nd013/15	553	79	0.8571
nd070/10	637	47	0.9262
nd070/12	864	122	0.8588
nd070/20	971	198	0.7961

7.3 Snakes

A thorough evaluation of the Snakes plugin with artificial data and a comparison to results from an alternative implementation is available in Karin Bokelmann's thesis [98]. Furthermore, sample segmentations of real image data of different organisms are presented in there to give an overview of the application spectrum of the plugin.

7.4 Conclusions on Quality

The presented data on cell detection and tracking quality provides an idea of the possibilities of the image analysis pipeline in contexts of real data. For common image qualities, the correctness of cell detection results is greater than $90\,\%$, as are the tracking results. Hence, the presented pipeline enables the investigation of biological questions with high-throughput MSCCs. Furthermore, the presented results suggest that an imaging setup has to be planned and adapted with the automated image analysis in mind.

8 Visualization and Analysis of Lineages

The following chapter is based on the publication "*Vizardous*: Interactive Analysis of Microbial Populations with Single Cell Resolution" [129].

SCA together with time-lapse (fluorescence) microscopy has been widely established in the life sciences in recent years [15, 52, 53]. Whereas flow cytometry yields population level data at a particular point in time, time-lapse live-cell imaging targets the assessment of individual cell parameters in their spatial and temporal (development) context. Microfluidic LOC technologies have enabled the highly parallel cultivation of hundreds of cells over several generations [2, 130, 131]. Combined with time-lapse imaging, it has been used to study the dynamics of cellular regulation mechanisms, to investigate, among others, cellular size homeostasis [60, 132] and phenotypic heterogeneity in dependence of stressors as well as the detection of rare events [133, 134].

The easy availability of time-resolved data poses new challenges on data analysis by the mere amounts and complexity of acquired information. While several, typically organism-specific, image analysis tools are available for cell detection, feature extraction, and tracking (for review see: [105]), tools for the generic task of analyzing and understanding this information are lacking.

Hence, we have developed the software tool *Vizardous* to assist researchers with the following single cell data related tasks: *a*) visualize data with single cell resolution in the lineage context, *b*) visually detect emerging structural patterns such as symmetries in subtrees, and *c*) assessing joint cellular property and structural motifs. A major contribution to the implementation has been made by Charaf E. Azzouzi in his internship as well as his Master thesis on "An interactive visualization and analysis tool for similarity testing of lineage trees".

8.1 Description of *Vizardous*

The single cell community has adopted phylogenies, to visualize mother-daughter relationships in context of cellular development and to link time-resolved data to the

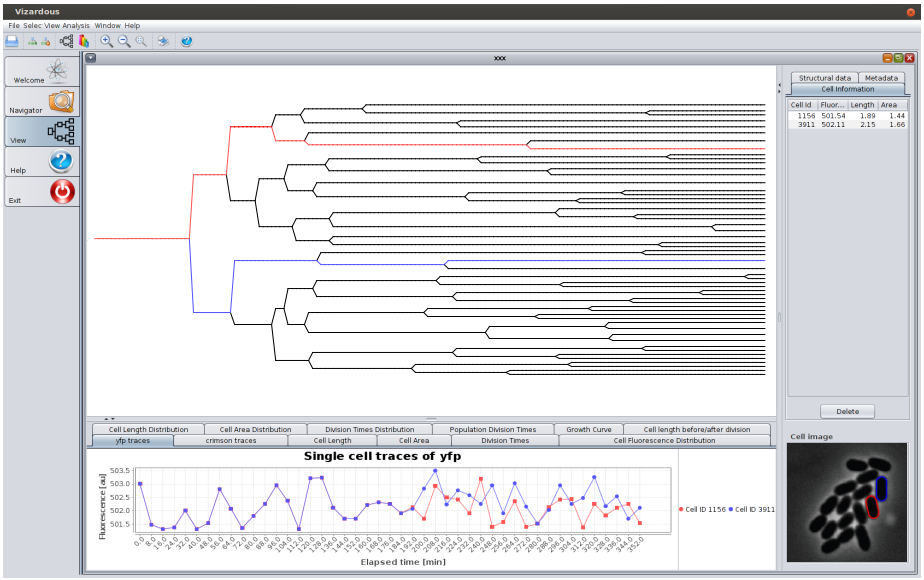


Figure 8.1: Screenshot of *Vizardous* user interface. The depicted lineage tree is taken from a nutrient limitation experiment. The chart at the bottom is zoomed in and, hence, just shows the beginning of the experiment.

tree structures. The use case for SCA is a different when compared to evolutionary biology. In SCA, lineage trees denote the connection between different "instances" of a cell across various frames in an image sequence.

Hence, a flexible "pattern and outlier" detection based on spatio-temporal phenotypic observables of individual cells is essential. Because the objectives in single cell analysis are notably different from those in evolutionary biology, conventional software tools for analyzing phylogenies are unsuitable in the single cell domain.

Vizardous is an open-source software that is designed as out-of-the-box solution to deal with the data from live-cell imaging experiments. The software is implemented using published methods and available libraries wherever appropriate (see Table B.1). Mother-daughter relationships are provided in the standardized phyloXML format [135] and cellular characteristics (e.g., cell length, area, volume, fluorescence reporter signal) as well as population average information in the specially designed document format MetaXML, respectively (cf. Subsection 8.3.2).

The intuitive user interface enables experimentalists to browse quantitative single cell data in the lineage context (cf. Figure 8.1). While the major use case of *Vizardous* is

the visualization of lineage trees, the underlying information can be also acquired from the data models using an API (cf. Figure 8.3) which allows users with programming experience to script the data analysis tasks directly with Java, Jython [136], or any other data analysis tool by using the available export capabilities. The next section will present a typical analysis workflow using *Vizardous*. Within this workflow, further interesting features will be highlighted.

8.2 *Vizardous* Analysis Workflow for SCA

The most important analysis steps are explained with a representative evaluation workflow using preprocessed time-lapse videos of a growing *C. glutamicum* population under transient carbon limitation (for details see Subsection 12.1.6 and Figure E.6):

Step 1: Importing Experimental Data Single cell data are read from phyloXML and MetaXML files pairs containing lineage information and figures about individual cells, respectively. One or more of such file pairs can be imported from a local file system or an external storage location, e.g., OMERO [137].

Step 2: Visualization of Lineage Trees Lineage trees are generated and visualized alongside population distributions for a complete experiment or a specific time point (temporal slice). As an indicator of population heterogeneity statistical moments (mean, SD) are calculated for the selection.

Step 3: Highlighting Individual Cells When screening for rare events or outliers in populations, the highlighting of single cells in a lineage tree according to cellular properties is especially useful. Three different approaches are implemented:

1. Change the property of a lineage tree element (node or branch) according to a specified cellular property, e.g., nodes are sized according to the cell area.
2. Set a threshold for a cellular property and highlight all cells that exceed the defined value.
3. Sort nodes and complete trees according to cellular properties (note that a total sorting of leaves might not be possible because of structural constrains, cf. Section 8.4)

Step 4: Selecting Cells for Inspection The user has the possibility to interactively select (and deselect) individual cells of interest from the lineage tree. The associated information is visualized in trace charts Figure 8.1. These charts unlock the contextual interpretation of feature dynamics along the temporal axis therewith bridging the gap between the ancestral relationships of cells and their meta-information.

Step 5: Exporting Lineage Trees and Charts The lineage trees and single cell traces are fully customizable and exportable to publication ready vector graphics (SVG) or bitmap formats (JPEG, PNG). Additionally, the data underlying single cell traces can be saved to tabular data formats (XLS, CSV) for further processing.

8.3 The *Vizardous* Data Model for Lineage Trees

The mother-daughter relationships of lineage trees as well as cellular information are read from a pair of XML files. Lineage information is stored in the well established PhyloXML format [135]. Cellular information and information that is not directly attachable to edges or nodes in a lineage tree are loaded from a new file format MetaXML. The development of a new format was necessary due to the lack of an established solution in the single-cell community.

8.3.1 PhyloXML

PhyloXML is a format that has been established by Han and Zmasek [135]. It is an XML-based, extensible file format for the storing of phylogenetic trees (mainly in the context of evolutionary biology) and their associated information. We have based our tool on PhyloXML to be able to use already available software for visualizing generated lineage trees from PhyloXML data although without meta-information. We refer the technically interested reader to [135] for further insights and the original XML schema definition.

8.3.2 MetaXML

While PhyloXML is very flexible and extensible, it is a purely graph centered file format. In general, information can only be attached to the elements of a graph, more precisely nodes and edges. This, however, hinders the integration of information that cannot directly be attached to such entities modeled by nodes (cells) or edges (mother-daughter relationships). A prominent example is information about an individual frame of an image sequence, e.g., background fluorescence or information about the population of cells as a whole, e.g., the center point of a population.

Therefore, we have developed a new, XML-based format that stores additional information regardless of their connection to cells. Following common naming schemes for markup languages, we have dubbed this format Meta-information Extensible Markup Language (MetaXML). The complete definition as XML Schema Definition (XSD) is supplemented as well (cf. metaXML-2.7.0.xsd). In the latest version of MetaXML (2.7.0 as of June 2015) the following cellular and non-cellular features are integrated:

- Experimental information

- Cell length per frame
- Cell area per frame
- (Approximated) Cell volume per frame
- Fluorescence reporters per cell per frame
  - Mean
  - Standard deviation
- Population center per frame
- Population area per frame
- Background / average fluorescence per frame

This list is continuously extended to incorporate additional information. An abstract description of the structure and the data in MetaXML files is provided in Figure 8.2. In-detail explanations of the file format are provided in Section B.3. An exemplary MetaXML file is listed in Listing B.1.

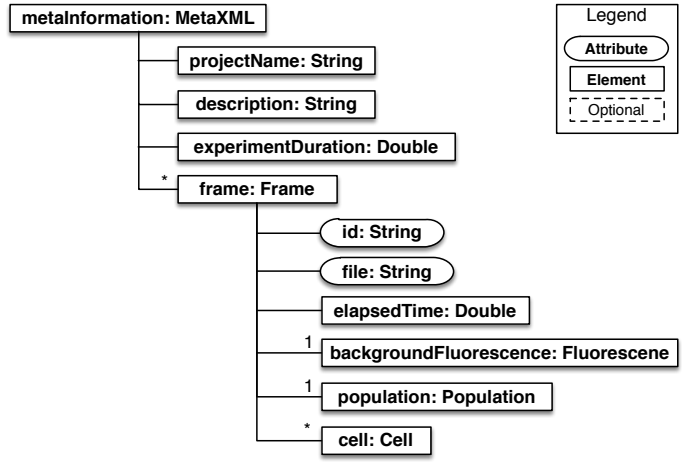


Figure 8.2: Schematic overview of the MetaXML format and the defined tags. Details on the different tags are provided in Section B.3.

8.3.3 The Internal Data Model of *Vizardous*

During the parsing of the previously described file formats, an internal data model is filled with the extracted information bridging the gap between the two file formats. This is necessary because the information that is stored in separate files describes the same biological entities that are accessed during a common (scripted) workflow.

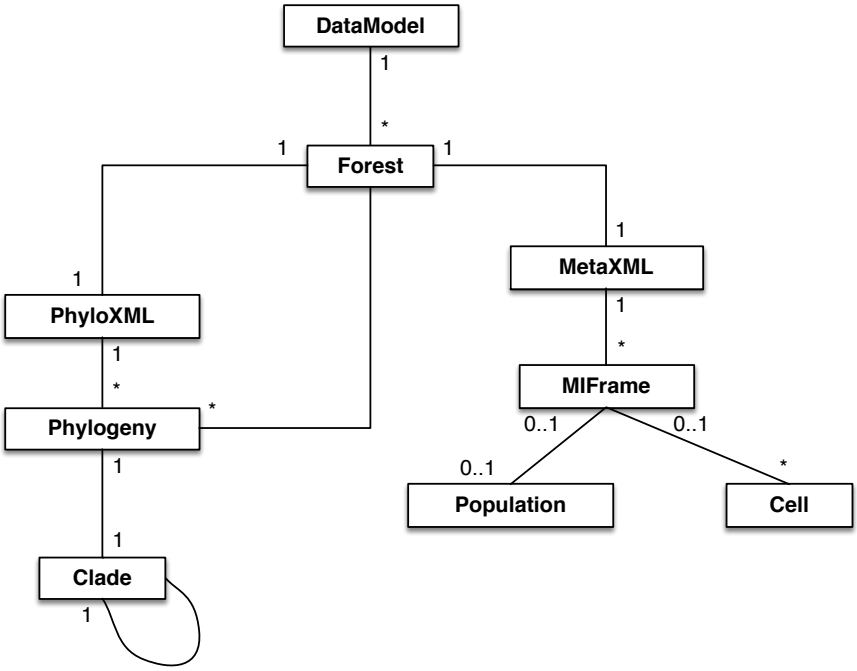


Figure 8.3: Simplified UML diagram of *Vizardous*' internal data model. Classes represent the structure of PhyloXML as well as MetaXML where appropriate. The Forest and DataModel classes bridge the gap between the structural and the cellular information. A DataModel can hold multiple Forests for comparative analyses of different experiments.

Figure 8.3 shows the connection between the main classes of the data model. The user will create a `Forest` from the pair of files. Subsequently, each `Phylogeny` that can be accessed from a `Forest` denotes an individual lineage tree for an individual cell at the beginning of an experiment. `Phylogeny` objects can be used to iterate the lineage tree in various ways using `Iterator` instances. It is also possible to directly gain access to the leaves of a lineage tree for deep assessment of those.

There are additional helper classes that model the data as stored in the file formats, e.g., `Cell` and `MIFrame`. Those can be used for accessing information with a cellular context (e.g. `Cell.getX()`).

8.4 Sorting Lineage Trees

One of the first steps after importing a PhyloXML/MetaXML pair is to identify cells of interest. What the characteristics of such a cell are and how "interesting" is defined differs from investigation to investigation. In many cases, for instance, outliers are of particular interest. An outlier usually is defined by a higher than usual or average value of a given cellular characteristic, e.g., cell length or fluorescence intensity. In order to identify such outliers, while still being able to visually compare the outliers property with the rest of a population, a lineage tree (i.e., its leaves) may be sorted such that cells with high (or low) values are at one end of the lineage tree, with low (or high) cells being on the opposite side of the tree. Hence, the possibility of sorting trees augments the highlighting of cells according to some fixed threshold.

In general, a sorting can be achieved for every cellular characteristic that has an implicit ordering. This means, that two cells have to be comparable. While this is true for two quantitative properties at any given time-point, taking into account the history and fate of a cell is challenging. For instance, taking into account a window of the last five time-points of a given characteristic. The mean value for this window is directly comparable, but "quantifying" the patterns in this window for a comparison is non-trivial.

As a working student, Johannes Seiffarth has implemented several sorting modes in *Vizardous* for solving the aforementioned challenges:

Subtree Sorting. In this mode, two different methods of sorting have been implemented. First, it is possible to sort a tree by considering cells with two children and bringing the two children in order. Using this approach, the subtrees that are rooted at these children are sorted implicitly. Second, it is possible to explicitly take into account values from the subtrees when looking at cells with more than one child. The general concept that was implemented is to compute a mean value for the subtree rooted at each of the children and sort according to the computed mean value. This approach is flexible in how a single quantity is extracted from the set of successors or leaves of the cell that is considered. While this is not available yet, it is trivial to implement a sorting that is, for instance, based on the standard deviation of all successor cells.

Leaves Sorting. In leaves sorting mode the goal is to only sort leaves according to a specified property. The previously mentioned remarks also apply to this sorting mode. Currently implemented are the trivial comparisons of directly accessible cellular quantities. However note, that it is not necessarily possible to sort the leaves due to limitations that are imposed by the structure of the lineage tree.

This limitation surfaces when two leaves that should be located next to each other according to the sorting of a cellular property do not have a common ancestor apart from the root cell. In this case, the leaves are located in completely different subtrees

of the lineage tree. Hence, it is not possible to obtain a complete sorting of the leaves without having edges of the lineage tree overlap.

With this limitation in mind, we are actively investigating possibilities to overcome the issue. One possibility is to use an alternative visualization scheme for mother-daughter relationships that relaxes the structural constraints on edges. Tree maps are such an alternative visualization to lineage trees with a more flexible way to visualize assignments. Figure 8.4 shows an example of a tree map.

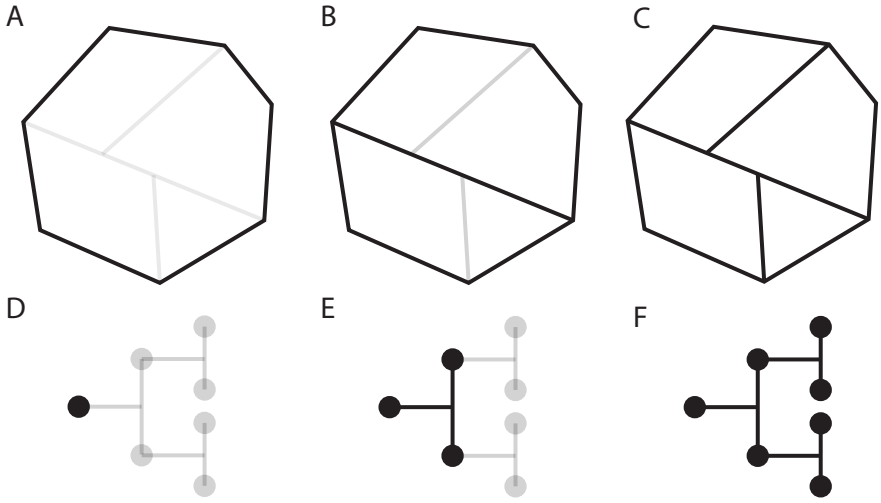


Figure 8.4: Tree map example. **A. - C.** Tree maps that are corresponding to different parts of the tree in D. - F. The area of the different parts denote the magnitude of an arbitrary cellular characteristic. **D. - F.** According lineage trees to the tree maps.

The general concept of a tree map is to denote mother-daughter relationships via inclusion in areas of the tree map. In the artificial example of Figure 8.4, the area of one contour might denote a specified cellular characteristic, for instance cell area. Hence, starting with the outer contour of the tree map, the enclosed area will contain all leaves of the lineage tree. At this stage, however, the area denotes the area of the root cell (as visualized in Figure 8.4D). When progressing in the experiment, Figure 8.4B shows the comparable area of the two daughter cells and so on. The advantage of this visualization technique is the addition of an additional dimension that can be used for a better position according to a sorting scheme.

8.5 Comparing Lineage Trees

Since cultivations are executed in a parallel manner in microfluidic LOC devices, lineage trees from such cultivations might be comparable. While such comparisons might dismissed as artificial, the motivation behind them is obvious. While sorting lineage trees can be used for finding individual outlier cells, comparing complete lineage trees can in practice be used to find outlier cultivations in a complete set of lineage trees from an experiment.

Hence, the comparison of lineage trees is motivated by two similar questions:

1. Are two or multiple trees similar with respect to a specific characteristic?
2. Is a lineage tree more or less heterogeneous than another lineage tree with respect to a specific characteristic?

While the first question focuses on the general problem of the similarity of two trees, the second question opens up a different perspective of similarity: heterogeneity. Although heterogeneity in a microbial populations is a complex topic in itself, a first concept to approach is, is to look at the similarity of subtrees within a cultivation.

These questions show that a comparison of either complete trees or subtrees is inherently coupled to the underlying property for which they are compared. Therefore, a comparison algorithm as well as a meaningful property that describes the specific research question adequately is required. Both will be described in the following sections.

8.5.1 Tree Comparison Properties

As previously mentioned, the properties or measures that are used for comparing trees are very specific to the data at hand. The cases can be roughly divided into two categories: structural data and cellular properties, where both categories are compared separately.

However, if one only compares structural data, cellular properties are completely discarded. This will effectively result in an analysis of cellular growth (patterns) since growth characteristics are encoded in the structure of a lineage tree. Approaches for this kind of comparison are described in Subsection 8.5.2 in more detail.

If only cellular characteristics are taken into account for comparison, the developmental context of a cell is completely neglected. In this case, for instance, it is only possible to compare the distributions of characteristics for cells or subsets of cells. Possible comparison properties are the mean value or standard deviation of the leaves of trees.

Combining both comparison approaches opens up a new field of research. This challenge can be tackled, for instance, by using available structural information for iterating trees and combining cellular properties in a recursive fashion. Such properties, however, have to be tailored to very specific use cases and investigations. Nonetheless, such metrics are of great interest for the classification of cultivations based on structural as well as meta information.

8.5.2 Tree Comparison Algorithms

Phylogenetic trees from evolutionary biology usually comprise labeled leaves from a fixed alphabet (i.e., species or sequence alignments). Hence, phylogenetic trees with the same set of leaves can be compared by transforming one tree into another using a predefined set of operations, e.g., insertion, deletion, renaming. The objectives of this approach are to find the *minimal* set of operations that transforms one tree into another by overlaying concordant labels. The size of this set (i.e., the number of required operations) is called the Tree Edit Distance (TED) [138].

Comparing lineage trees extracted from single-cell experiments with this approach is, however, not possible. This is due to the fact that we are comparing lineage trees of different alphabets because individual cells are never part of two cultivations. Hence, the aforementioned TED algorithm is not applicable for the data from MSCC.

In addition, the original TED algorithm adheres to the ordering of trees which is not given in single-cell data, although trees might be ordered as presented in the previous section. This ordering is, however, artificial and user-generated. In general, two lineage trees are assumed to be equal irrespective of the ordering of children of cells with multiple children.

Therefore, we are looking to compare unlabeled, unsorted, "mostly" binary trees. Trees are "mostly" binary because only in rare cases a bacterial cell will split into more than two cells. This might not be true when analyzing algae or fungi. In our case, however, misassignments are a result of errors in the tracking process. While solutions exist for comparing labeled trees [138], they are not applicable for main use-case.

There is, however, an application for tree comparisons in a rather different field: automated comparison of the fragmentation patterns of small molecules in mass spectrometry. Hufsky *et al.* have developed a dynamic program approach for aligning unordered, unlabeled mass spec trees [139]. Since there is no implementation readily available, we have not tested the approach yet with data from MSCCs.

The previously mentioned approaches compute a single value for a pair of trees. Hence, to compare several trees with each other requires the computation of pairwise metrics. We are currently investigation alternative approaches for one-to-many comparisons. The differences between the concepts are depicted in Figure 8.5. One-to-many comparisons

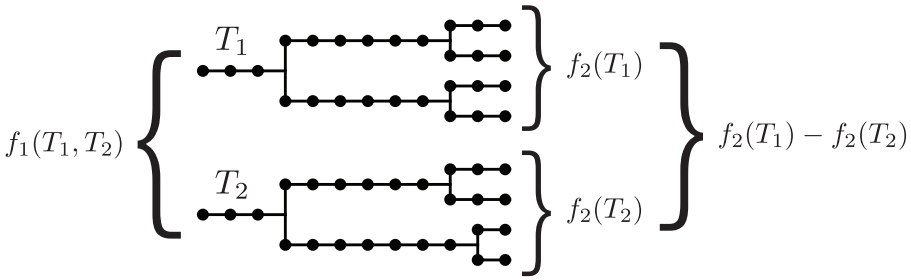


Figure 8.5: Schematic of one-to-one approaches and one-to-many approaches. On-to-one approaches are shown on left side, where one comparison metric $f_1(T_1, T_2)$ is established. The concept of one-to-many approaches is to compute a metric f_2 for each tree individually and subsequently compare the computed values.

enable, among other use cases, the clustering and with that the comparison of multiple trees with a reduced computation cost. On a more general note, such approaches are required for finding patterns in the comparison values in a larger set of lineage trees.

8.6 Results and Conclusions

Vizardous has been used for the analysis of data of the studies in Part III. Apart from that, *Vizardous* has been used in several studies, e.g., [26, 46, 140, 141], by scientists with different backgrounds to answer a variety of scientific questions. These studies show that *Vizardous* is a versatile tool that supports researchers in various aspects with the discovery-driven analysis of single cell experiments. By supporting additional input formats and seamless integration with available bioimage analysis software we anticipate that a range of fields including biological and medical sciences will benefit.

Part III

Applications

Overview

Microfluidic single-cell cultivation combined with time-lapse microscopy opens up a plethora of possible applications [53]. In the following, a short and concise overview of the types of experiments that are possible with SCA and are executed in a biotechnological context at IBG-1 is provided.

- Characterizing microfluidic LOC devices
  - Readouts:
    1. Growth rate for a whole population
    2. Cell position in the growth site
    3. Cell position with respect to the population
- Investigating bacterial growth
  - High temporal resolution to keep uncertainties in the reconstructed lineage tree to a minimum
  - Rather few MGCs because of the high temporal resolution (experimental/technical limitation)
  - Readouts:
    1. Growth rate for a whole population (if it is based on cell number then we do not need a lineage tree: less error-prone)
    2. Distribution of division times
    3. Elongation rates for individual cells
    4. Cell position with respect to the population
- Investigations of (intra)cellular processes
  - Population patterns are of interest (formations of subpopulations)
  - Outliers / rare events are of interest
  - In case of rare events: many MGCs to sufficiently sample rare events

- Readouts: Time-resolved fluorescences intensities for up to 3 (usually 2) fluorescence channels

The following chapters will focus on specific applications and case studies from the aforementioned fields. While most of the applications are driven by a biological question, there are also applications that are driven by the technological advances. This is on the one hand the optimization of LOC designs, on the other hand the image analysis pipeline.

9 Device Characterization

A main motivation for research on microfluidic LOC devices is their broad range of applications in the field of bioprocess development [15, 25]. This is derived from the fact that microfluidic chips can provide a constant cellular environment in contrast to large scale bioreactors [142]. In order for microfluidic LOCs to be used as tools by the process engineering and biotechnology community, devices have to be thoroughly characterized.

During the development of microfluidic LOC devices, chip designs have to be repeatedly characterized using different technologies. The set of technologies spans computational fluid dynamics (CFD) simulations [143] as well as analyses based on fluorescence-labeled latex beads that are used to visualize the flow through the chip [61]. These approaches are a first step in the direction of a "*[. . .] better understanding of the mass transport within the microfluidic chip systems [. . .]*" [144].

Such characterization approaches have in common, that they omit the biological system and the influence of the chip on the organism. In order to analyze these influences and investigate the assumption of constant nutrient supply, advanced image analysis techniques are required for cell identification and data processing. Hence, pipelines like the one presented in Part II are used for characterizing devices in the context of biological applicability.

9.1 Nutrient Limitation in Growth Sites

One major goal of the device characterization approaches is to show that cells in a microfluidic growth site are not nutrient-limited, regardless of their location in the population and the growth site. First investigations based on CFD simulations with uptake modeling have been conducted by Westerwalbesloh *et al.* [143]. Figure 9.1 shows the population that has been modeled in an MGC and the glucose concentration at constant flow through the chip. The authors suggest from their simulations "that the cells in the middle of the center of the large colony reach uptake rates above of $97\,\%$ of the ones in pure growth medium."

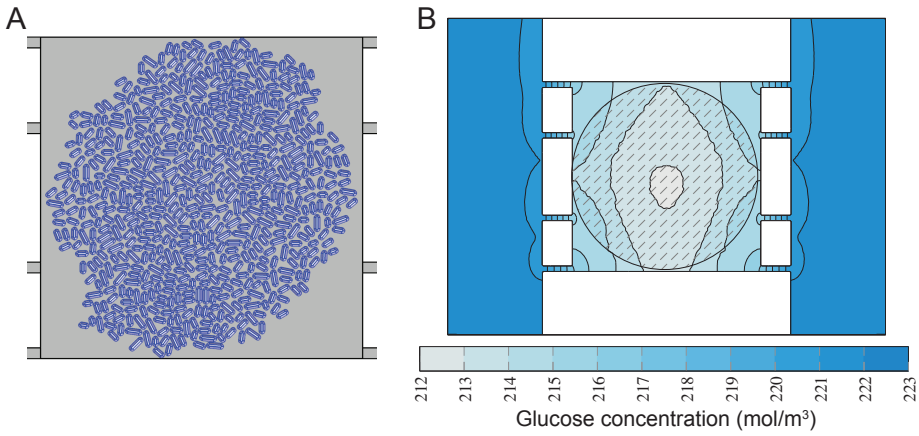


Figure 9.1: CFD simulation results for MGC. **A.** Model of a large colony extracted from an exemplary microscopic image of a real colony in a MGC. **B.** Glucose concentrations in an MGC. The striped area indicates the position of the bacterial colony (adapted from [143]).

In order to investigate nutrient limitation effects under real conditions we are using the developed image analysis pipeline to extract morphological parameters, i.e., cell area, for individual cells. To that end we are computing the center of mass for each individual cell in a population together with the population center. Judging from previous experiments and from the images that are shown in the left column of Figure 9.3, a correlation between the parameters seems reasonable.

This is, however not the case, as the right column of Figure 9.3 suggests. The last frame of the image sequence is particularly interesting for this investigation. Figure 9.2 shows an enlarged version of the scatter plot together with a regression using a first as well as a second order polynomial. The first order polynomial does not capture the relation between the properties with a reasonable confidence. The second order polynomial seems a better fit for the data but lacks a biological interpretation.

One hypothesis is that cells near the perimeter of a population are consuming the nutrients so fast, that at some point during an experiment cells are not growing anymore. Hence, the data might better be fitted with a biphasic regression model. The MGC used for the experiment has dimensions of $80\,\mu\mathrm{m} \times 80\,\mu\mathrm{m}$. Hence, the nutrient supply might indeed be a problem in comparison with the MGC designs that are $40\,\mu\mathrm{m} \times 40\,\mu\mathrm{m}$.

Hence, this example is rather of artificial nature, but shows the limitations of microfluidic LOCs that can be investigated using image analysis techniques. The image analysis pipeline can complement and confirm previously published simulation studies.

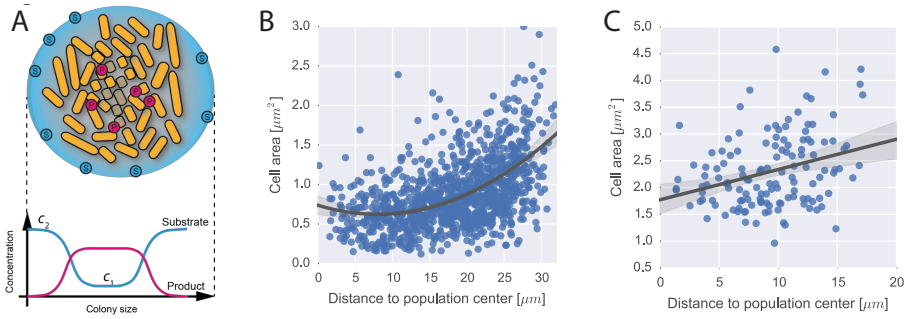


Figure 9.2: Effect of nutrient gradients on cells in a population. **A.** Schematic of nutrient limitation in a MGC (adapted from [144]). **B.** Scatter plot of cell area over distance to the population center. Data taken from the last frame of Figure 9.3 with a second order polynomial fit to the data. **C.** Correlation between cell area and distance to population center for a single cultivation (complete data are presented in Chapter 10).

9.2 Optimizing Inoculation Procedures for Single Bacteria

Microfluidic cultivation devices can contain many different geometries for bacterial cultivation. 2D MGCs where cells are not able to overlap and grow into monolayer populations; 1D growth channels where cells are trapped in such a way that they only grow in one direction; 0D traps that are only able to hold one individual cell at a time [145]. All the aforementioned geometries have different characteristics [15].

Since a considerable number of applications for microfluidics are concerned with population or cellular heterogeneity, the objective is to investigate clonal populations. To that end, it is beneficial that a MGC is only filled with an individual cell at the beginning of a cultivation. If that is, however, not the case, it is still possible to decouple the development of cells using image analysis techniques. Then, the assumption that each cell in the cultivation chamber stems from one cell (i.e., they shares the same genetic origin) might be violated in such cases.

Up until recently, the seeding procedure (i.e., the filling of MGCs with bacteria) was based solely based on diffusion. Recently, Probst *et al.* [61] have proposed a new procedure to seed bacteria into the cultivation chamber. The general concept of the method is the insertion of an air bubble into the microfluidic chip, which transiently changes the laminar flow through the chip in such a way, that cells are flushed into the growth sites. This is in strong contrast to the previous procedure, where mass transport was diffusion-based because of equal flow in parallel media channels.

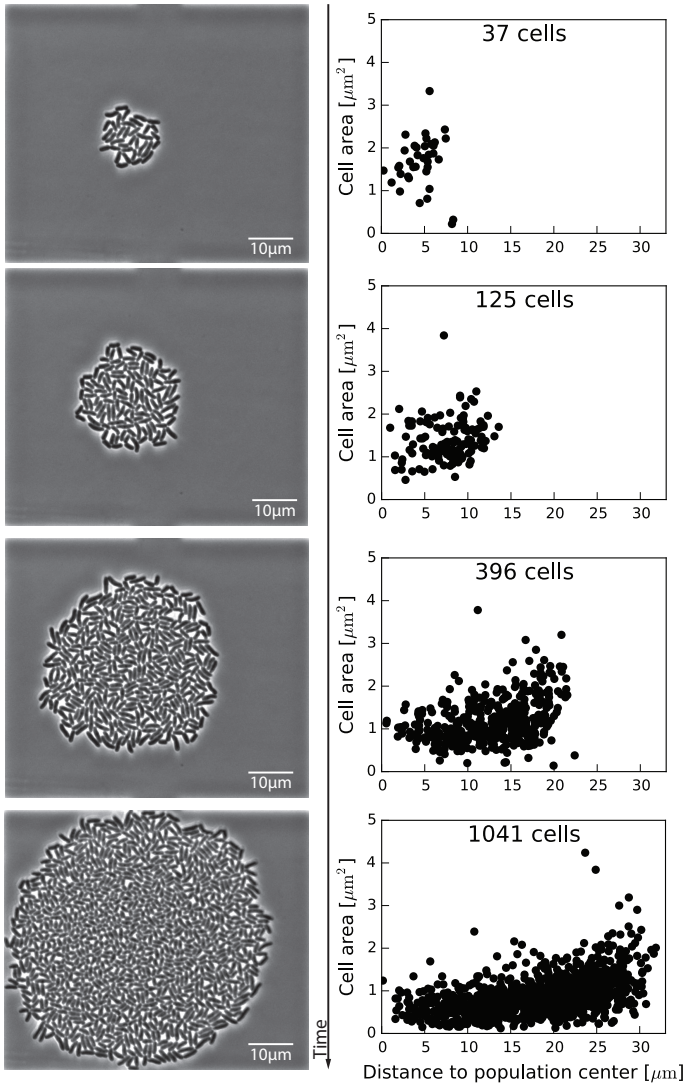


Figure 9.3: Analysis of microbial growth with respect to the position in a population grown in a monolayer growth chamber. The left column shows a sequence of images from a cultivation experiment. At the end of the sequence, it seems that cells near the population center are smaller in comparison to the outside. The right column shows plots of cell area vs. the distance to the population center for individual cells. The first frames barely show a correlation. At higher cell densities a linear correlation might be devised.

In order to prove the applicability of the procedure and find the optimal experimental parameters (geometry size and optical cell density of seeding suspension) the presented image analysis pipeline has been applied. To that end, four parameter sets have been evaluated in a comprehensive cell trapping analysis. This analysis features the number of cells per cultivation chamber and the spatial location of cells in a MGC. Chambers have been categorized according to their number of seeded cells where chambers without cells are also taken into account to compute a chip-wide seeding efficiency Figure 9.4.

The location of cells is of increased interest, because cells that are located near exits are more likely to leave the growth site during an experiment. In most cases, the reason for a single cell leaving a growth site is the flow through the cultivation chamber. In addition, populations that are developing from cells near exits are prone to growing into the exits and out of the region of observation. The computational analysis of populations with cells near a boundary or an exit of a growth site is more demanding than exposed cells in the center of an MGC.

9.3 Conclusions

While microfluidic LOC devices are on the advance, the characterization of such devices is not exhaustive until the biological specimen is taken into account. With automated image analysis techniques (e.g., the image analysis pipeline proposed in this thesis) this gap can be diminished if not closed. Especially, the dependency of microbial growth on the location in the MGC is worth further investigations. First results from which a slight correlation might be extracted have been presented, if only for a subpopulation.

On a second note, automated image analysis has been used to improve the seeding procedure for microfluidic chips. To be as close as possible to reality, we have used bacteria in the seeding tests instead of beads. This way, the peculiarities of bacteria are already taken into account with the evaluation method.

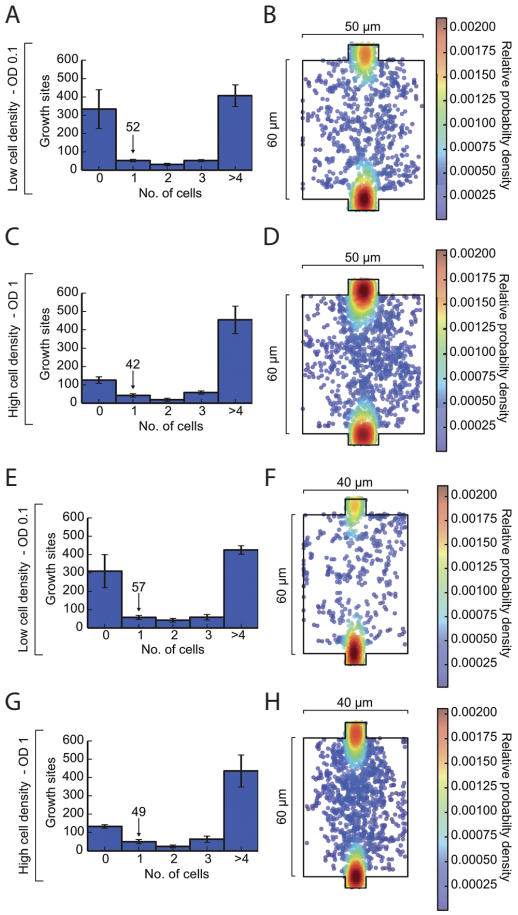


Figure 9.4: Cell trapping analysis ($N = 3$) of *C. glutamicum* at $300\,\mathrm{mbar}$ and different cell suspension densities for growth sites of $50\,\mu\mathrm{m} \times 60\,\mu\mathrm{m}$ ($N = 592$) and $40\,\mu\mathrm{m} \times 60\,\mu\mathrm{m}$ ($N = 592$). (a and b, e and f) OD 0.1: inoculation efficiency distribution and single-cell location plot; (c and d, g and h) OD 1: inoculation efficiency distribution and single-cell location plot (adapted from [61]).

10 Quantifying Bacterial Growth

This chapter will focus on the question of how to evaluate single-cell cultivations in the context of bacterial growth. Before the fact that modeling approaches for microbial growth in bulk experiments are not necessarily applicable to MSCCs is highlighted, an introduction on microbial growth in general will be presented. Subsection 10.1.1 focuses on the general (i.e., not organism specific) question of how to extract growth rates for MSCC. The underlying data for this investigation have been acquired for *C. glutamicum* cultivations.

In contrast, Section 10.2 contains results on the investigation of the individual, single-cell growth of *C. glutamicum*. At the core of this section is the question of the underlying growth model for individual *C. glutamicum* cells.

The chapter concludes with some hints and best practices on how to evaluate single-cell experiments with comparability in mind. While the focus is here on the comparability between MSCCs on LOC platform, comparability with bulk measurements is discussed as well.

10.1 Population Growth

Growth and division of individual cells are fundamental for all living organisms, ranging from humans to bacteria. Gaining an increased understanding of bacterial growth and division patterns is indispensable for many fundamental and applied research fields. The applications range from understanding and developing novel antibiotics against bacterial pathogens [146] to improvements in bacterial production processes of industrially relevant bulk chemicals [147]. Especially investigations of growth-coupled production processes have been executed in batch and chemostat cultivations of varying scale for many decades. To determine colony responses to process parameter changes, different bulk methods including cell counting and cell density measures have been used.

Under conditions where nutrients are limiting, the growth of a bacterial population can be divided into several phases according to Monod [148]: lag, exponential, stationary, death. A microbial growth curve (i.e., cell density over time) of this shape can be described with a variety of phenomenological models (see [149] for review). Among others, it can be modeled with a logistic (Verhulst) function. While also more sophisticated methods have been reported in literature [150], their applicability to data generated with microfluidic LOC devices is limited.

This is mostly due to the fact, that LOC platforms have been developed to provide a constant cellular environment and are operated under conditions that are not limiting growth. In this cultivation mode, cells are constantly kept in the exponential growth phase [59, 151]. Thus, growth models that describe the classical multiple growth phases model of Monod [148] cannot be applied when investigating growth of microbial populations in MGCs.

The most prominent parameter for bacterial growth, i.e., specific growth rate μ, can be extracted from experimental data using multi-phase growth models as well as simpler ones. The specific growth rate is unique for a bacterial species in conjunction with experimental parameters. Historically, μ refers to a constant in the exponential growth law [148]. Hence, the practical implementation of the computation of μ changes for limiting and non-limiting cultivations and their (non-exponential) growth curves.

In addition, bulk measurements average over a large numbers of cells and so they mask cell-to-cell variability in division times, sizes at division, growth rates, and other properties like single-cell production potential. In the last years microfluidic tools have increasingly been used for quantitative studies of these masked single-cell growth parameters [152–154].

Recently, different concepts of colony-based bioreactors on single-cell level have been published [59, 155]. They offer the possibility to investigate complete colonies, comparable to large-scale experiments (cf. Figure 10.1), but still provide the chance to probe individual cells under constant environmental conditions.

Hence, these technologies allow to also address more fundamental questions of growth and division at single-cell level:

- How can we quantify bacterial growth processes in a robust and comparable way?
- How comparable is a growth phenotype from a macroscopic (e.g. growth rate of a population) and microscopic (e.g. cell division rate of a single cell) perspective?

To answer those questions, we have comprehensively determined and compared the specific growth rates of *C. glutamicum* populations during microfluidic cultivation. The focus here is to use the advantages of MSCC, namely a high temporal resolution combined with the throughput of the method. We also consider the fact, that the experiments are challenging as well as the extraction and evaluation of acquired data.

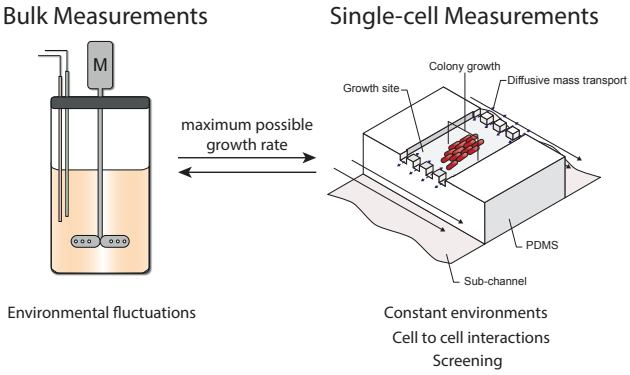


Figure 10.1: Comparison of large-scale and micro-scale cultivation techniques.

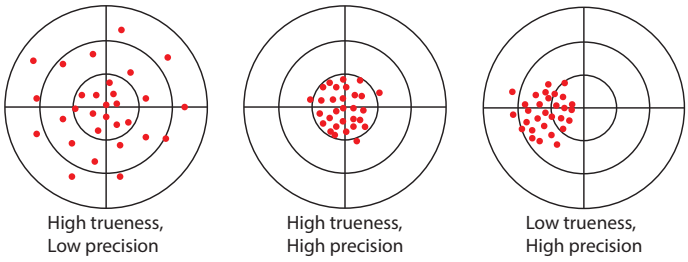


Figure 10.2: Schematic of measurement accuracy. The center of the board denotes the true value in each case. Red points are read-outs using measurement techniques with different accuracy characteristics. Adapted from [156].

10.1.1 Quantification of Population Growth in SCA

We are looking for a quantification method of the specific growth rate in MSCCs that complies with the maximum specific growth rate as extracted from batch cultivations. In addition to this objective, the quantification method should have a high trueness (replicate the underlying, true value) as well as a high precision (accurately determine the true value). These two objectives, as well as other undesired outcomes are shown in Figure 10.2. A method with low precision but high trueness is usually biased by a random effect and is, therefore, hard to optimize. A low trueness with a high precision, in contrast, is a challenge inherent to the measurement method and might be solved by improving the method itself. First, however, such systematic errors have to be uncovered which is barely possible, if measures are not comparable with available methods.

The bacterial growth rate is defined as the increment ratio of a population, which is

measured by the increment in cell mass or in viable cell number. Since bacteria grow in a monolayer on the LOC devices made in Jülich, the sum of cell masses in a colony is considered to be directly proportional to the population area. The population area, however, could be extracted with more crude measures that do not integrate knowledge about individual cells.

In literature, several methods for determining growth rates for microbial colonies are presented [16, 36]. We have summarized the available methods in Figure 10.3.

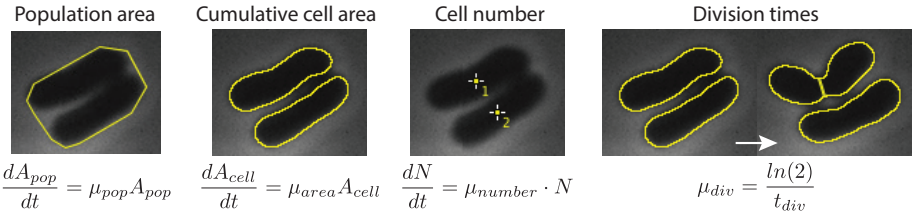


Figure 10.3: Different quantification methods.

10.1.2 Comparing Quantification Methods

In order to systematically compare the previously described quantification methods for growth rates, we have cultivated *C. glutamicum* in MGCs using complex BHI medium (cf. Section C.1). During the cultivation, images were acquired every two minutes ($\Delta t = 2\,\mathrm{min}$), compared every $5\,\mathrm{min}$ to $15\,\mathrm{min}$ in typical experiments. While the high imaging frequency does not improve accuracy of "spatial measures" (areas and cell numbers), division times t_{div} for individual cells are determined more accurately with a higher acquisition frequency. The division time is defined as the time between two cellular division events. Hence, the improvement is due to the fact that when a cell division is detected, it has actually happened in between the last frame and the frame of detection (i.e., $t_{div\_detected}$). The division time and the time between two images Δt are, therefore, related by

$$t_{div} = (t_{div\_detected} - \frac{\Delta t}{2}) \pm \frac{\Delta t}{2}.$$

We have computed growth rates for five selected cultivations using the aforementioned quantification methods. The resulting growth rates using the area-based measures were $\mu_{area} = 0.75 \pm 0.01\,\mathrm{h}^{-1}$ and $\mu_{pop} = 0.74 \pm 0.02\,\mathrm{h}^{-1}$ for the cumulative cell area and the population area, respectively. In contrast to that, using the cell number as basis for the computation yields a specific growth rate of $\mu_{number} = 0.81 \pm 0.01\,\mathrm{h}^{-1}$. A side-by-side comparison of the cell number plot as well as the cumulative cell area over time with the respective exponential fits is shown in Figure 10.4.

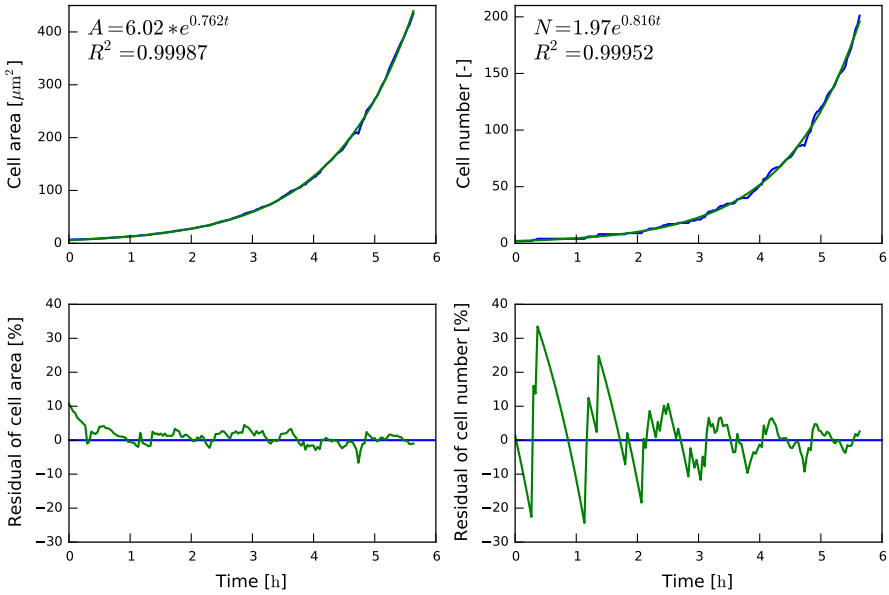


Figure 10.4: Comparison between an exponential fit to the cumulative cell area of a population and the cell number of the same population (DatasetID 35/1). The blue line denotes measurement data, while the green line is the fit evaluated at the same positions as measurement data. The inlays show the fitted function with the optimal parameters and the R^2 value of the fit. The lower row shows residual plots for both quantification methods. The blue line denotes the exponential fit, the green line shows the deviation from experimental data in percent.

While the difference in the computed growth rates is about $7.5\,\%$ ($\mu_{number} > \mu_{area}$), both measures almost perfectly fit to an exponential growth model. When looking closer at the underlying fits, the R^2 values are comparable. Hence, we have decided to take a closer look at the fits with residual plots (cf. Figure 10.4 bottom row). These plots reveal the fact, that especially at lower cell densities, the quantification based on cell number performs poorly in comparison to using the cumulative cell are. Hence, we suggest that one has to determine cellular growth via cell area when focusing on the beginning of an experiment where the cell density is comparably low. Furthermore, we confirm that when taking into account a broader temporal context and higher cell densities, both measures lead to comparable results.

From lineage trees it is possible to deduce exact division times with a high precision, as already described in the first paragraph of this section. Therefore, we have also reconstructed trees for the same five cultivations. Using the mean value of the division

time distribution that is shown in Figure 10.5, we have computed

$$\mu_{div} = \frac{\ln(2)}{t_{mean}} = 0.80 \pm 0.01\,\mathrm{h}^{-1}.$$

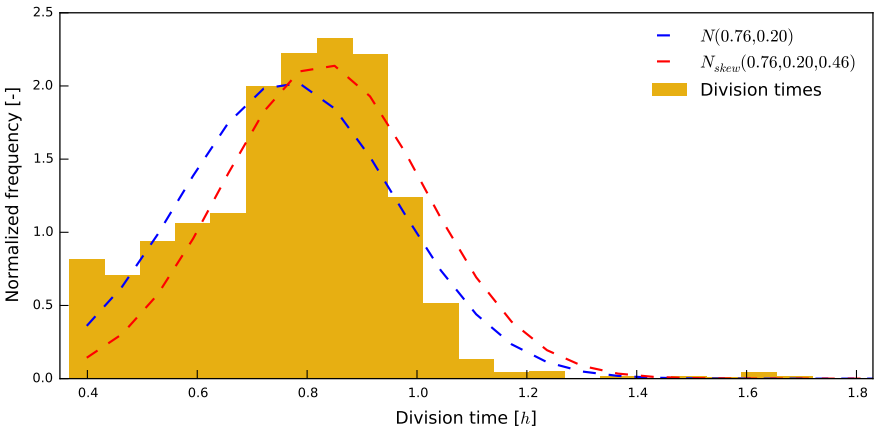


Figure 10.5: Histogram of division times computed from various cultivations ($n = 5$).

In order for this computation to be accurate, the average cell length has to converge to a constant value for cell number $N \to \infty$. At the core of the problem is the case where the average cell length decreases during an experiment while the division time remains constant. In this case, the division time can not be used as a proxy for the biomass increase over time anymore. This is not the case for our experimental dataset, the resulting data is visualized in Figure 10.6.

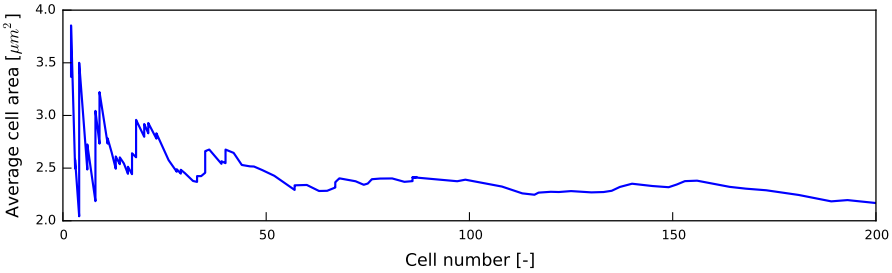


Figure 10.6: Average cell length converges with increasing number of cells.

All in all, Figure 10.7 sketches the possible quantification methods and their connection. With respect to the objective of finding a method with high trueness and high

precision, we cannot elicit a general statement. This heavily depends on which measurement is used as a reference for the "true" value (Figure 10.7 uses μ_{div} as reference). In the end, the decision for a quantification method has to be based on the experimental data (short term or long term) as well as the information that is desired. Hence, we extend the conclusion drawn by Dusny *et al.*: "Consequently, growth rate determination on the basis of cell volume or mass measurement is clearly superior to mere cell number or elongation rate determination [...]", when in the regime of low cell density [54].

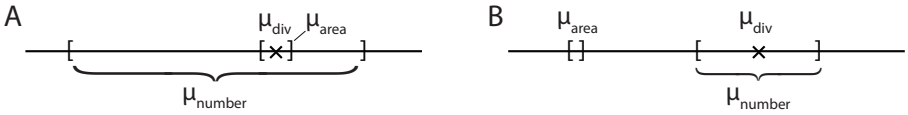


Figure 10.7: Precision of the different quantification methods. **A.** At the beginning of an experiment. **B.** At high cell densities, when individual cell cycles are asynchronous.

10.2 Single-Cell Growth

The previous section has described general methods for the quantification of microbial growth, i.e., specific growth rates. The presented quantification approaches are neither organism specific nor do they directly incorporate single-cell growth. In that regard, these measures are comparable to bulk measurements from large-scale cultivations like measures of optical density.

However, to investigate single-cell growth, microfluidics or other SCA techniques are required. With the wealth of information such techniques provide and the availability of automated image analysis, the underlying mechanism and growth model of individual cells can be investigated. Furthermore, *single-cell growth rates*, per convention called *elongation rates* in the following, might be a factor worth considering for future optimization of bioprocesses and production strains. While the specific growth rate of a population is a good indicator for microbial growth, it covers growth heterogeneity of individual cells. For example, the growth properties of a strain that produces a slow growing subpopulation during cultivation offers room for improvement of the slow growing population on the single-cell level.

Knowledge about the growth of individual cells has been generated for decades [157]. Two competing hypotheses, linear or exponential growth, have been tested for *C. glutamicum* for the first time by Dusny *et al.* [36]. This in strong contrast to the available data sets and literature about *E. coli* in that field [14, 60, 132, 158, 159]. Hence, we have evaluated a dataset with a high temporal resolution not only for the population growth rates but also cell area traces for individual cells. Figure 10.8 shows all extracted cell area traces of the dataset.

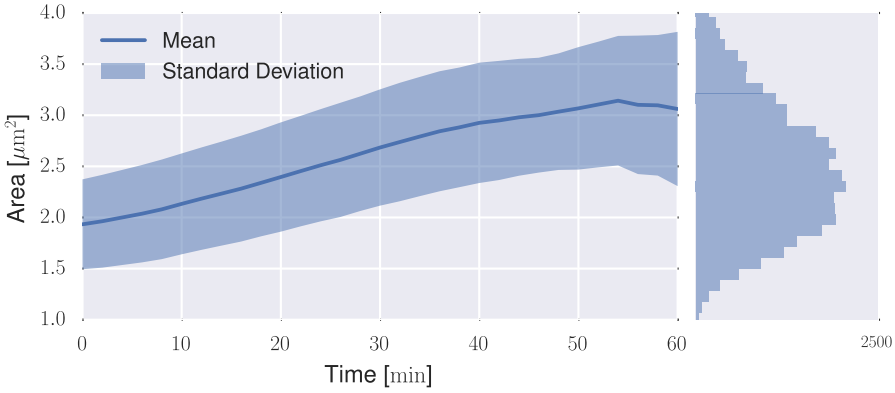


Figure 10.8: Traces of cell area for a selected number of cells ($n \approx 1200$) from lineage trees. The first generation of an image sequence as well as the last generation till the end of an experiment have been discarded. The histogram on the right shows the distribution of cell areas irrespective of time.

We have decided to use the cell area as a proxy for determining the traditionally used parameter of cell volume or biomass. The decision is based on the fact that *C. glutamicum* is rod-shaped and keeps its cell width constant during a generation. Therefore, the connection between the elongation rate from cell area and the one extracted from cell length (data not shown) is described as:

$$E_{area} = \frac{\mathrm{d}A}{\mathrm{d}t} = \frac{\mathrm{d}(L \cdot W)}{\mathrm{d}t} \qquad \text{with } W \text{ const.} = W \cdot \frac{\mathrm{d}L}{\mathrm{d}t} = W \cdot E_{length}. \tag{10.1}$$

To show how the two proposed growth models, i.e., linear and exponential, perform on single-cell area traces of *C. glutamicum* cultivations, Figure 10.9 highlights several traces alongside linear and exponential fits.

10.2.1 Linear or Exponential Growth Model of *C. glutamicum* cells?

While Figure 10.9 only shows a few, selected traces, we have evaluated hundreds more individual cell traces of their respective cell area. In order to determine the goodness of fit, we have computed R^2 for both linear and exponential fits for each extracted trace. The results from the study are presented in Figure 10.10.

The general impression is, that a majority of the fits are located in the upper right corner of the plot. Points in that particular area have high R^2 values for both growth models. In contrast to that high density region of data points, a trend seems to emerge

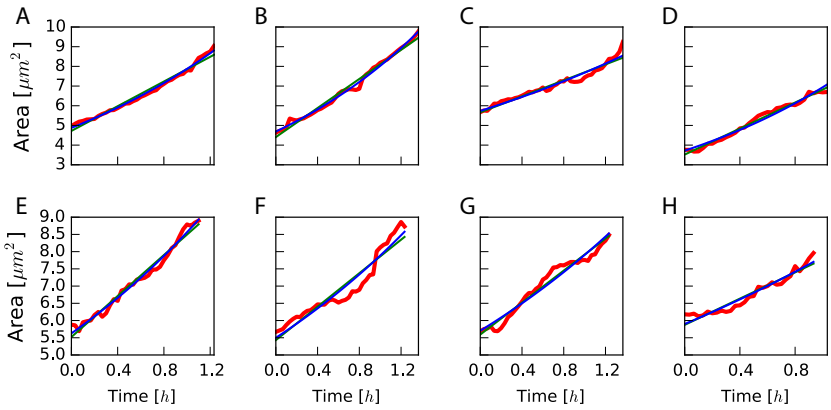


Figure 10.9: Selected cell area traces. Red lines show the measurement data, green lines a linear fit, and blue lines an exponential fit.

for fits where both R^2 are suboptimal. Visual inspections give the impression, that exponential fits performs better in this regime. Nevertheless, the impact of this trend is low because the deviations between the two models are less that $10\,\%$ in all cases (as illustrated by the band around the $R^2_{linear} = R^2_{exp}$ line).

In cases where the data is very noise, e.g., Figure 10.9F, neither model shows preferable results when compared to each other. Since both growth models feature two parameters, i.e., the cell area at the beginning of a generation and the increase over time, we cannot determine which model to select based on *Occam's razor*. The principle of *Occam's razor* states that if two competing hypothesis perform equally well, then one with less parameters or fewer assumptions should be selected.

In the top right area of the scatter plot, which is magnified in the inlet of Figure 10.10, a trend is hard to elicit. To resolve that issue, we have computed the count of fits above and below the equality line of the two growth models, i.e., linear and exponential. There are 757 values over the equality line in contrast to 531 below the line, while most of the fits show less than $5\,\%$ deviation from each other (all fits are in the $10\,\%$ band).

With this data at hand, it is hard to come up with a biologically motivated conclusion about whether growth of individual *C. glutamicum* cells follows a linear or an exponential model. For *C. glutamicum* cultivated in BHI medium, there seems to be hardly a difference between the two competing hypotheses. While the presented data renders an exponential growth model more likely, the number of fits where a linear models is more appropriate is not negligible.

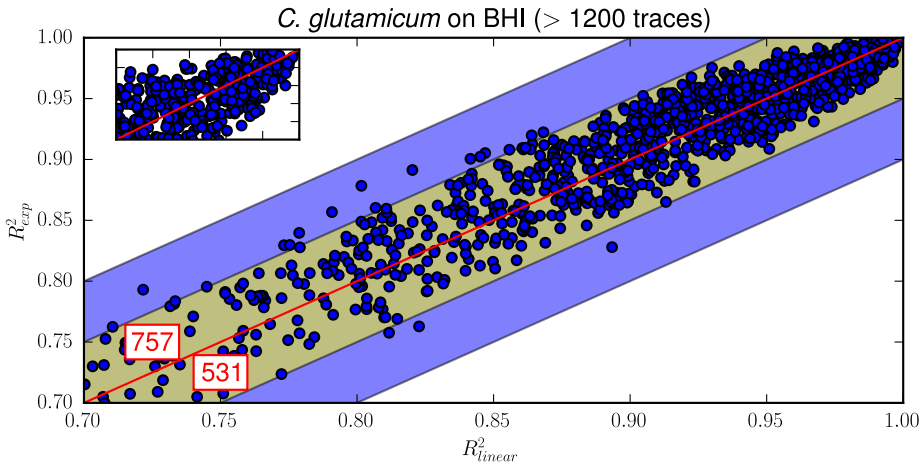


Figure 10.10: Goodness of fit compared for linear and exponential model. The yellow and blue band is the 5 % and 10 % deviation from the equality line, respectively.

Recently published literature [60, 132, 160] has focused on growth investigations with *E. coli* and *Bacillus subtilis* in microfluidic devices. The presented data originates from mother machine data, where individual cells can be observed for prolonged periods of time. The length of individual cells is an easily accessible and robust growth parameter [60], that is barely influenced by the cellular neighborhood and image analysis artifacts.

With *C. glutamicum* cultivated in MGCs, the accuracy of cell detection for the extraction of cell length or cell area might still shadow the underlying growth model. For the presented data, we deem both models applicable due to the minimal deviations from one another.

10.3 Conclusions

We have showed in a systematic study, that especially at the beginning of an experiment the cell number is not a reasonable indicator for microbial growth. Hence, we conclude that one has to determine specific growth rates from cumulative cell area at low densities. At higher cell densities, we confirm that both measures lead to comparable results, where the computation of cell area at high cell densities is biased.

In contrast to the exponential growth of bacterial populations, the single-cell growth study has revealed that a clear distinction between linear and exponential growth cannot be derived. Since the growth of individual cells is determined by the metabolic state of a cell, data for different medium compositions is required. In Chapter 11 we apply the presented quantification methods to a dataset with different medium compositions.

However, in this study we have focused on complex BHI medium, where we could neither accept nor decline the hypothesis of exponential single-cell growth.

All in all, if one is comparing single-cell growth measures with bulk approaches, it seems that the only viable approach is to extract division times for individual cells. This ansatz, however, comes with a major drawback: required analysis and computation times (cf. Figure 10.11). Therefore, the extraction of single-cell division times might be a very accurate measure for individual growth, but the advantages diminish, when looking at the time effort for construction of required lineage trees.

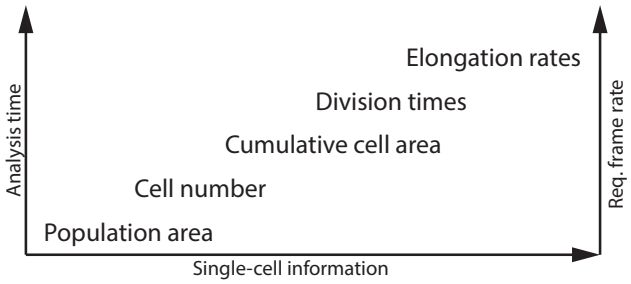


Figure 10.11: Schematic of the analysis and curation time for different growth quantification methods in comparison with the information content.

11 Morphology Screening

SCA enables the extraction of organism parameters that are of great interest for systems biology. By executing MSCC with an analysis on single-cell level, additional parameters for process engineering can be obtained. With SCA, instead of looking at population average information like maximum growth rate, it is possible to zoom in on individual cells. Correlations between parameters that might be covered by the averaging over a complete microbial population by bulk measurements can be resolved. This study demonstrates that a (semi-)automated image analysis pipeline is vital to take advantage of the potential of MSCC for the extraction of and the screening for systems biology relevant parameters, i.e., single-cell growth and morphology.

The cultivations for the data shown in this chapter have been executed in the context of Johanna Heinrich's Master's project. This project was supervised by Alexander Grünberger, who has presented preliminary results on population level in his PhD thesis [144].

11.1 Screening for Cell Morphology

When screening for particular cultivation parameters, e.g., medium composition, and their influence on bacteria, a high reproducibility of experiments is required. Especially in the context of medium composition, a plethora of additives and components have to be screened. Agar-pads that are usually used for the execution of such studies are not applicable if tens to hundreds of cultivation regimes have to be tested in short time under constant conditions. Microfluidic LOC devices offer the possibility to expose all cells to the same medium composition, in contrast to a shaking flask or lab-scale bioreactor. State-of-the-art time-lapse microscopy is used for the acquisition of image sequences of population development. This allows for visual parameter control and monitoring during the cultivation. Images are acquired at $15\,\mathrm{min}$ intervals resulting in image sequences of growing microbial populations.

In this case study we are investigating and characterizing the physiology of *C. glutamicum* in different environments. *C. glutamicum* is a model organism for the

biotechnological production of amino acids as well as higher value compounds [34]. Many studies have focused on metabolism and resolving cellular metabolic pathways [30, 161]. In addition, it is well known that *C. glutamicum* is able to grow on and use various carbon sources including glucose, fructose, sucrose, xylose, and arabinose [162]. Building on the available information, we are investigating the cellular response, i.e., growth and morphology patterns, to several key carbon sources of the glycolysis and the central carbon metabolism as an example for the screening of different environments.

Screening different carbon sources for their influence on population growth, cell elongation, and size regulation in *C. glutamicum* extends available literature on these topics [163–165] with regard to the set of nutrients.

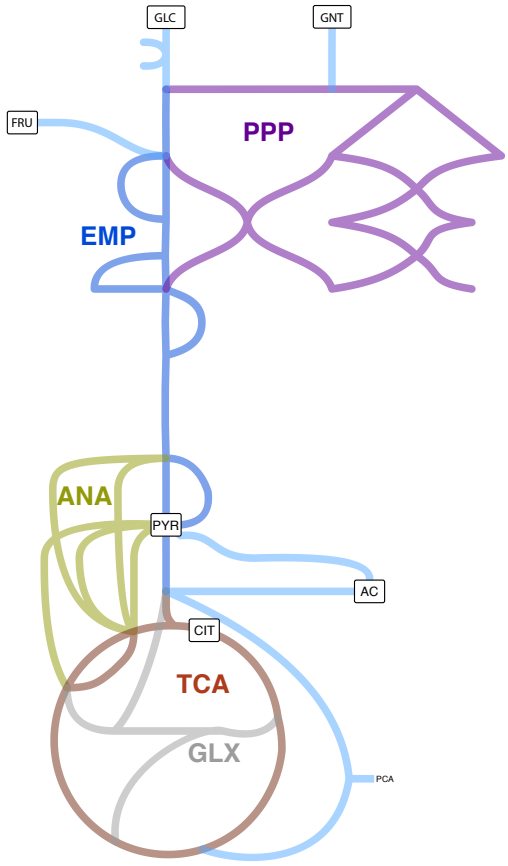


Figure 11.1: Schematic of the central carbon metabolism. Entry points of the studied carbon sources into the central carbon metabolism are highlighted with rectangles.

11.2 Results

MSCCs of *C. glutamicum* are carried out in well-defined minimal medium (CGXII [166]) with addition of various carbon sources, namely glucose, gluconate, fructose, pyruvate, acetate, citrate, as well as without one of these carbon sources. In the latter case, protocatechuic acid (PCA) is metabolized as sole carbon source that is also added in the other medium compositions for experimental reasons. The carbon sources of interest are close to the central carbon metabolism (cf. Figure 11.1). They are, however, located in different pathways, i.e., the glycolysis and the pentose phosphate pathway.

For these cultivations, we have extracted population-based growth rates, single-cell elongation rates, as well as morphological parameters for individual cells.

11.2.1 Population Growth

The population growth rates μ_{pop} of *C. glutamicum* populations have been determined under varying cultivation conditions. Subsequently, a subset of three MGCs has been evaluated such that the cell number for each frame can be extracted as well as cumulative cell area. This enabled the computation of the growth rate from both cell number and cumulative cell area according to the methods published by Grünberger *et al.* [2].

While we have applied different quantification methods to obtain maximum growth rates, Figure 11.2 shows the growth rate mean value and the standard deviation computed from the evaluated triplicates based on cell number. A closer look at the differences between the two quantification methods has been provided in Chapter 10.

Under default conditions (i.e., CGXII without glucose with PCA) a growth rate of $\mu = 0.10 \pm 0.01\,\mathrm{h}^{-1}$ was computed. A significantly higher growth rate was computed for pyruvate ($\mu_{\mathrm{PYR}} = 0.36 \pm 0.01\,\mathrm{h}^{-1}$), acetate ($\mu_{\mathrm{AC}} = 0.46 \pm 0.01\,\mathrm{h}^{-1}$), and glucose $\mu_{\mathrm{GLC}} = 0.51 \pm 0.02\,\mathrm{h}^{-1}$. A slightly increased growth rate was observed for fructose $\mu_{\mathrm{FRU}} = 0.55 \pm 0.01\,\mathrm{h}^{-1}$ and citrate $\mu_{\mathrm{CIT}} = 0.63 \pm 0.01\,\mathrm{h}^{-1}$. Cultivations with gluconate as carbon-source showed the highest growth rate of $\mu_{\mathrm{GNT}} = 0.71 \pm 0.03\,\mathrm{h}^{-1}$ of the minimal medium compositions. The highest overall growth rate, however, was computed for cultivations on BHI with $\mu_{\mathrm{BHI}} = 0.84 \pm 0.04\,\mathrm{h}^{-1}$. In addition, the last two mentioned cultivations show the highest variance in the set of compositions.

All in all, the values ranged from $\mu_{min} = 0.09\,\mathrm{h}^{-1}$ to $\mu_{max} = 0.88\,\mathrm{h}^{-1}$ for default conditions and BHI, respectively. It is also interesting to note, that there seems to be a trend for higher growth rates to have higher standard deviations.

11.2.2 Single-Cell Growth

When applying the complete image analysis pipeline and evaluating the available data to the full extent, it is possible to investigate parameters of growth and morphology for

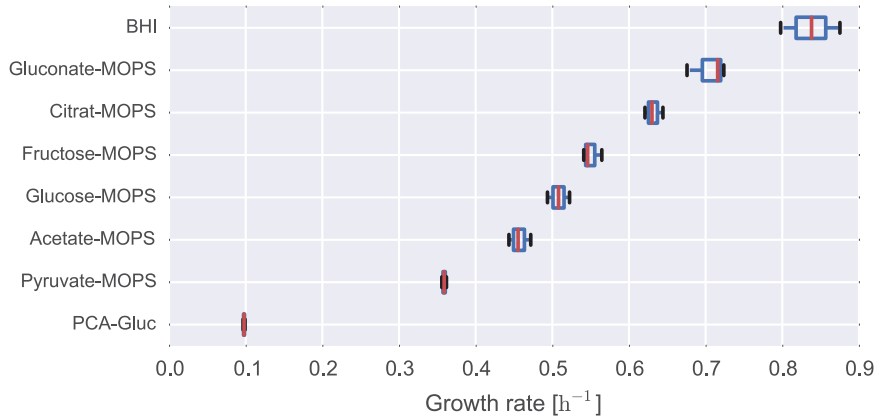


Figure 11.2: Population-based growth rates for different carbon sources. Carbon sources are sorted according to the median of their growth rates.

individual cells. Hence, we have computed cellular elongation rates as a read-out for single-cell growth as a proxy for the biomass increase over time (cf. Section 10.2).

We have extracted the elongation rates under the assumption that the cell area increases linearly over time for *C. glutamicum*. The question if a single bacterium grows exponentially or linearly is a long standing problem [167–169]. However, internal studies (cf. Chapter 10) have revealed that the differences in computed elongation rates are negligible. Therefore, we have executed a linear fit to the area trace of each cell, where the slope denotes the elongation rate. The data for the different carbon sources is depicted in Figure 11.3 with the sorting adapted to Figure 11.2.

Without the explicit addition of a carbon source, elongations rate of $E = 0.005 \pm 0.002$ $\mu m^2 min^{-1}$ were obtained. Cultivations on pyruvate and acetate resulted in elongation rates of $E_{PYR} = 0.010 \pm 0.003\,\mu m^2 min^{-1}$ and $E_{ACT} = 0.021 \pm 0.004\,sm^2 min^{-1}$, respectively. Addition of glucose and fructose showed elongation rates of $E_{GLC} = 0.020 \pm 0.007$ $\mu m^2 min^{-1}$ and $E_{FRU} = 0.020 \pm 0.005\,\mu m^2 min^{-1}$. MGCs with citrate and gluconate showed considerable higher elongation rates of $E_{CIT} = 0.028 \pm 0.008\,\mu m^2 min^{-1}$ and $E_{GNT} = 0.031 \pm 0.009\,\mu m^2 min^{-1}$. The highest elongation rate is computed for cultivations with BHI medium ($E_{BHI} = 0.050 \pm 0.010\,\mu m^2 min^{-1}$). The extracted elongation rates show the same trend that has already been described for the population growth. Namely, higher elongation rates also show higher standard deviations in our cultivations.

11.2.3 Single-Cell Morphology

Looking at individual cells of a cultivation/population, it becomes obvious that the morphology of cells changes with the different carbon sources. Building on previously

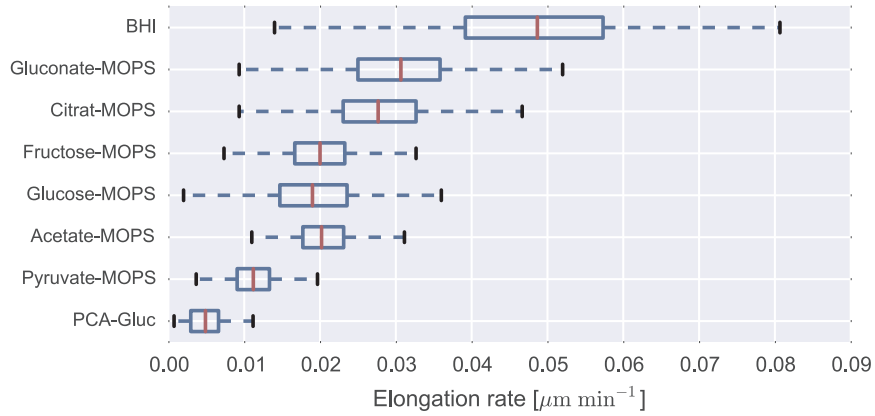


Figure 11.3: Single-cell elongation rates obtained for different carbon sources under the assumption of a linear elongation model. The plot is sorted according to Figure 11.2 for comparison.

observed cultivations, we have extracted cell length and cell area, respectively, before and after a cell division as well as the average of a cellular generation. Figure 11.4 and Figure 11.5 show the resulting distributions together with an excerpt from the image sequences (at higher cell densities) obtained by time-lapse microscopy.

On average, the smallest cells are observed when cultivations are conducted without additional carbon sources with $l = 1.76 \pm 0.23\,\mu\text{m}^2$. Cells cultivated with pyruvate are in the same order of magnitude: $l_{\text{PYR}} = 1.90 \pm 0.23\,\mu\text{m}^2$. Fructose and glucose show average cell lengths of $l_{\text{FRU}} = 2.49 \pm 0.29\,\mu\text{m}^2$ and $l_{\text{GLU}} = 2.45 \pm 0.37$ μm^2, respectively. The next group of carbon sources are about $20\,\%$ bigger with gluconate ($l_{\text{GNT}} = 2.89 \pm 0.37\,\mu\text{m}^2$), acetate ($l_{\text{ACT}} = 2.98 \pm 0.29\,\mu\text{m}^2$), and citrate ($l_{\text{CIT}} = 2.93 \pm 0.33\,\mu\text{m}^2$). The by far largest cells are observed for cultivations with BHI which show an average cell lengths of $l_{\text{BHI}} = 3.55 \pm 0.54\,\mu\text{m}^2$.

11.3 Discussion

We have analyzed MSCCs in LOC devices using the proposed (semi-)automatic image analysis pipeline to screen various cultivation conditions. Cultivations were done with different main carbon sources added to a well-defined minimal medium with the goal to investigate the impact on the growth and morphology of *C. glutamicum*.

To that end, we have extracted population growth rates for different carbon sources of the central metabolism of *C. glutamicum* over 5 to 6 generations. While we could not find a direct correlation between the entry point of the carbon source into the

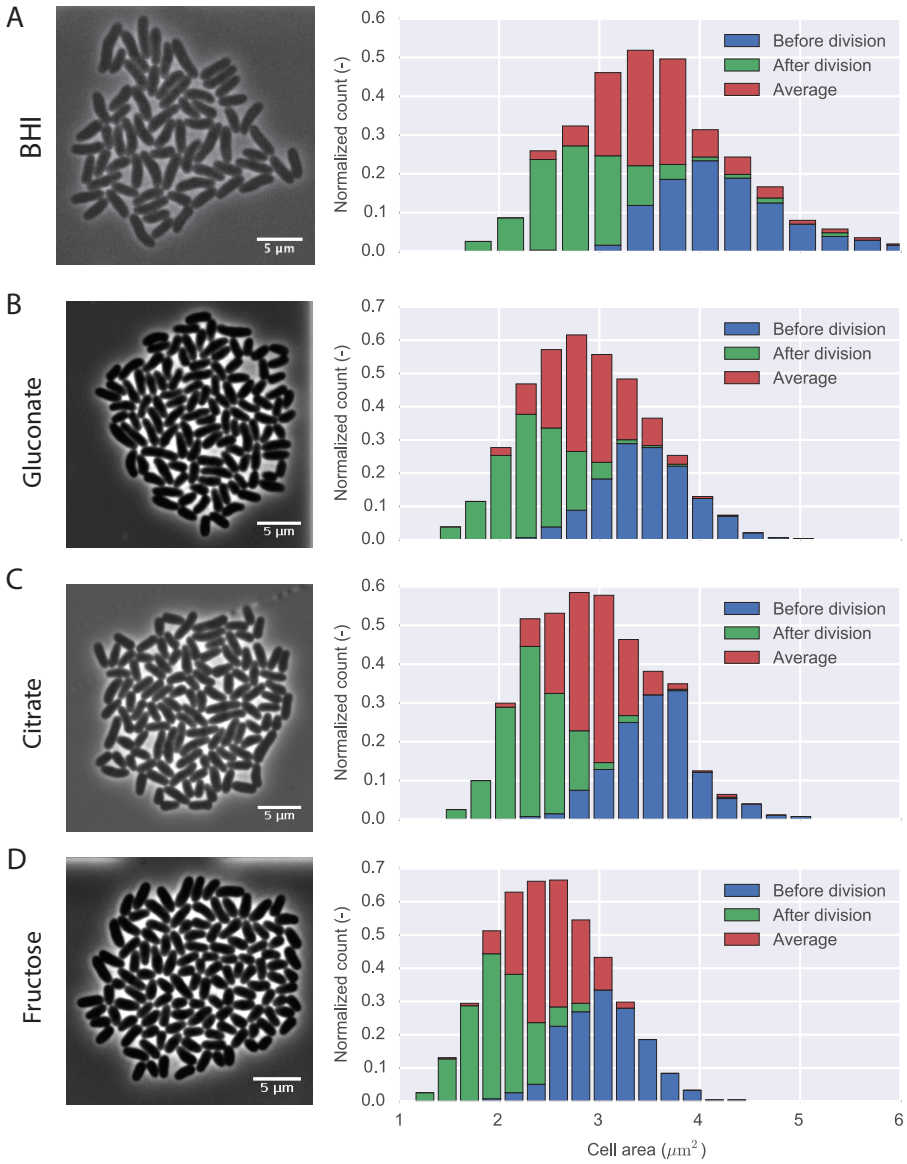


Figure 11.4: Morphology parameters for each carbon source. Histograms of cell area before and after a cell division.

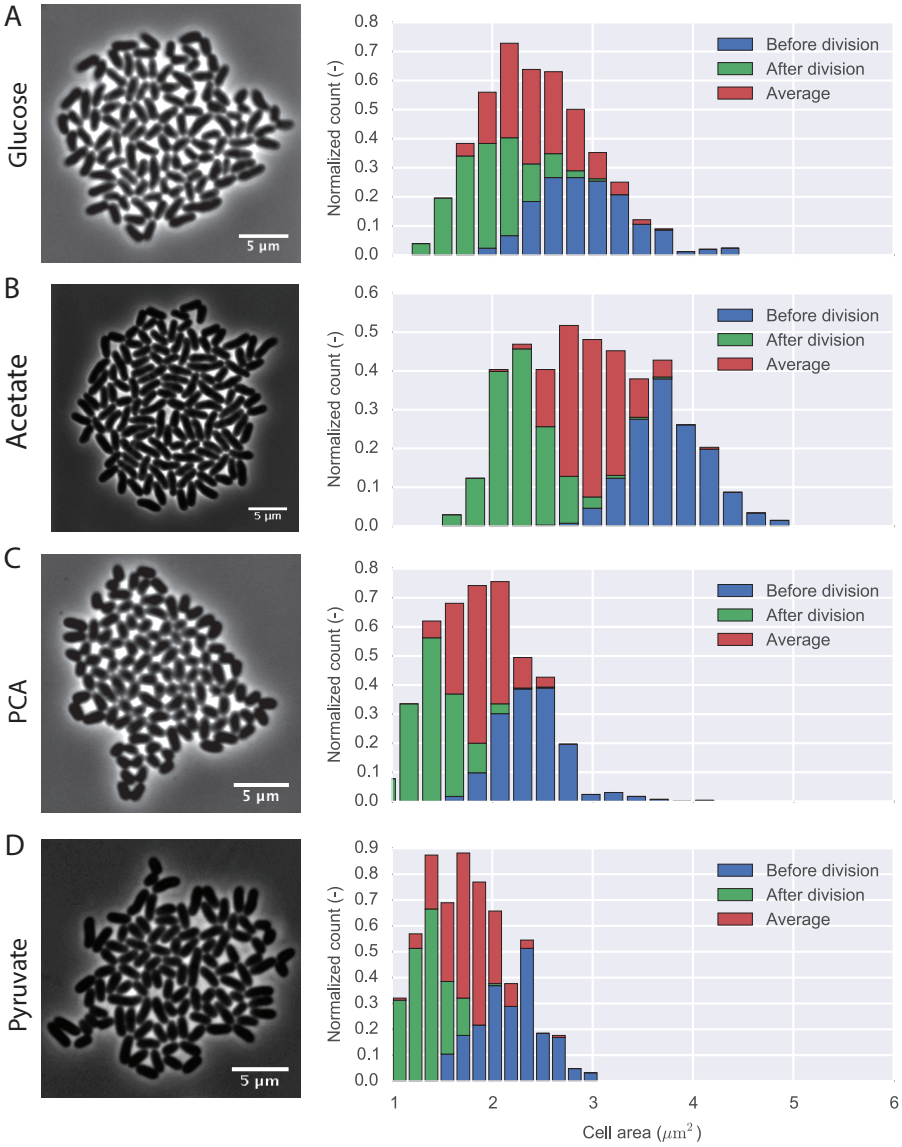


Figure 11.5: Morphology parameters for each carbon source. Histograms of cell area before and after a cell division.

central metabolism, the C-sources seem to cluster into different categories. Within these categories, the variance between the C-sources is negligible.

On the population level, growth rates show a clear correlation with morphological parameters for the investigated carbon sources (as depicted in Figure 11.6). Hence, the single-cell data followed established patterns known as the “growth law” [157, 170]: the average cell size increases as the growth rate μ increases. The same trend is observed when zooming in to single-cell elongation rates. In the end, population growth is governed by the growth of individual cells.

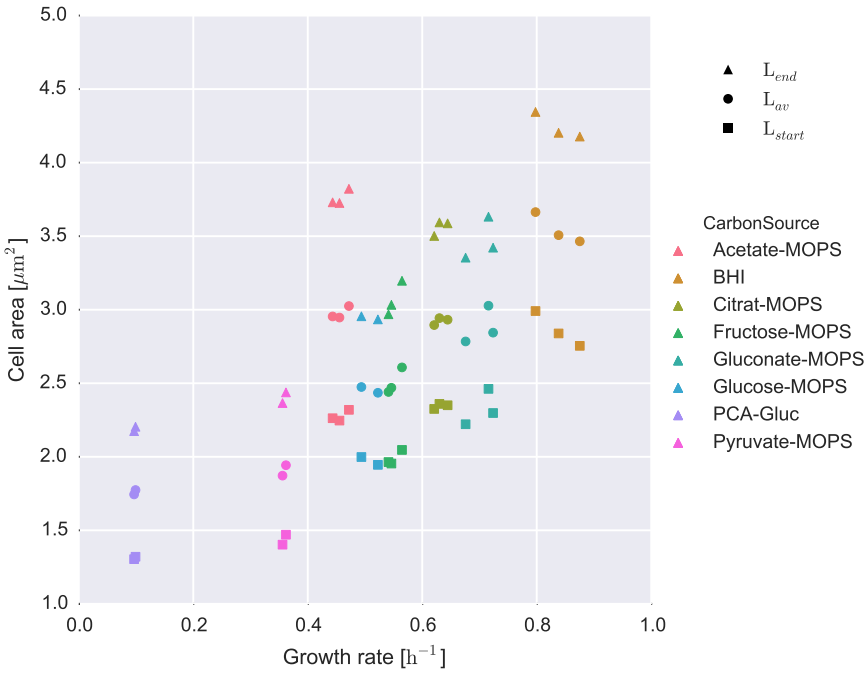


Figure 11.6: Correlation between population growth rate and morphological parameters for different carbon sources.

Taking into account morphological information for individual cells, it is possible to investigate the mechanism of size homeostasis. Figure 11.4 and Figure 11.5 show characteristic distributions of cell area for different carbon sources. The question remains, how these characteristic distributions are generated by individual cells and how this is orchestrated.

In literature, three competing models for size homeostasis are discussed [163]. A *timer*-based model assumes that the division time for individual cells remains constant

irrespective of the cell length at birth [171]. The second model, *sizer*, assumes a cell size threshold that has to be exceeded for a cell to divide [172]. A recently established hypothesis is the *adder* principle that states that cells add a constant factor Δ to their size and divide when this criterion is reached [160, 163, 173]. To answer the question which model is implemented by *C. glutamicum* we have derived various correlations between crucial growth and morphology parameters (Figure 11.7 and Appendix D).

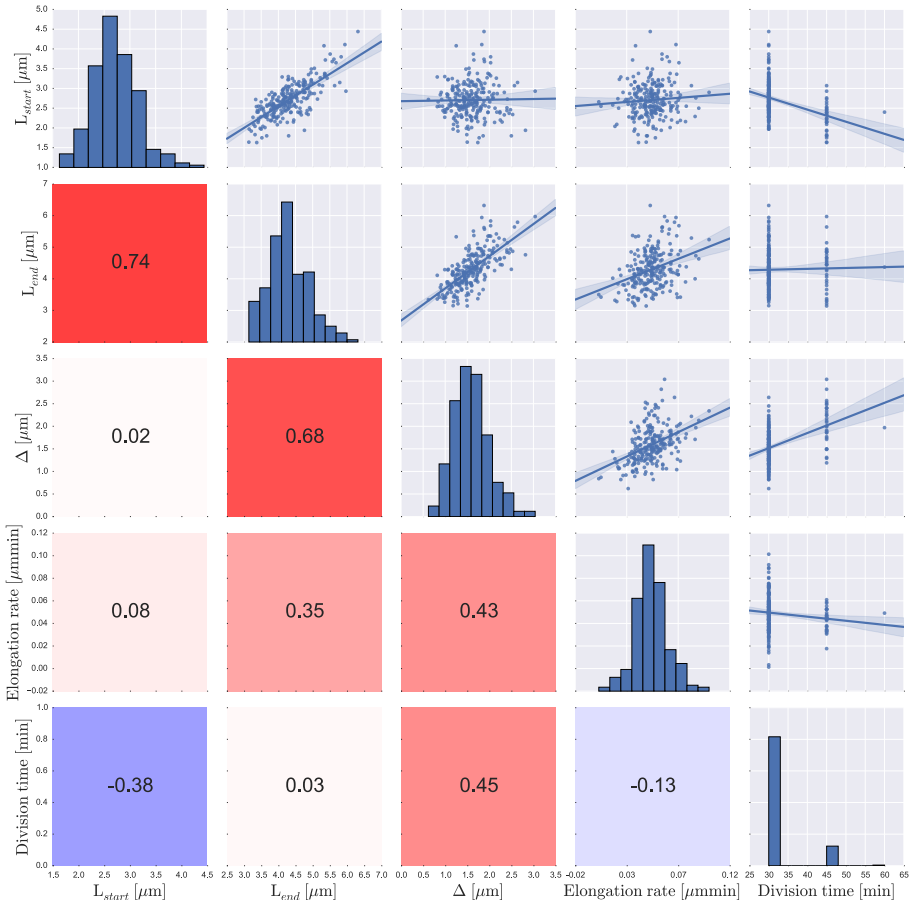


Figure 11.7: Connection between single-cell growth and cell morphology in a correlation plot for phenotypic features based on cultivations with **BHI medium**. The values in the lower half of the matrix denote the Pearson correlation coefficient. Histograms for each characteristic are shown on the diagonal. The upper block of the matrix shows scatter plots of pairs of characteristics with a regression line and the 95% confidence band of a linear model.

The negative correlation between the area of cells at the beginning of a generation

and the division time precludes the previously described *timer* model for cultivations with BHI medium. With our systemic screening of carbon sources we could show furthermore, that the Pearson correlation coefficients range from -0.56 to -0.38 for pyruvate and BHI cultivations, respectively (cf. Figure 11.7). Correlation plots for the remaining carbon sources are provided in Appendix D.

Furthermore, we cannot hypothesize a *sizer* model due to the high correlation between cell area before splitting and the cell area at birth. A size threshold as described by the *sizer* model would show a constant (up to stochastic fluctuations) cell size at the end of a generation irrespective of the size at birth. The correlation coefficients for our single-cell date range from 0.29 to 0.74 for pyruvate and BHI, respectively. In addition to ruling out a *sizer*-based model, the linear correlation suggests that cells add a specific size before dividing, according to the *adder* principle.

To investigate the applicability of this principle, $\Delta = \mathrm{L}_{end} - \mathrm{L}_{start}$ has been computed for the extracted single-cell information. According to literature, Δ remains approximately constant for cells of microbial model organisms, e.g., *E. coli* and *B. subtilis* [160, 163, 173]. Our data does not suggest a direct correlation between the parameters, hence, neither entirely supporting the *adder* hypothesis nor ruling it out.

In contrast to these correlations, there seems to be barely any correlation between the cell area at the beginning of a generation and the elongation rate over the course of the same generation. The highest correlation coefficient is computed with 0.39 for cultivations in bare CGXII medium (PCA as sole carbon source). However, the strong correlation between L_{start} and t_{div} suggests, that the size control is rather effected by adaptations in doubling time than the individual elongation rate. These findings are in consent with the work published by Osella *et al.* [159], presenting a size correction based on the modulation of doubling time.

11.4 Conclusion

In this study, we have used a microfluidic LOC platform for the parallel cultivation of *C. glutamicum* under varying conditions. We have cultivated bacteria in minimal (CGXII) medium that has been charged with different carbon sources to screen the space of media compositions.

In addition to looking at the growth of cells on population as well as on single-cell level, we have also extracted morphological parameters for individual cells. A connection between the average cell size and the population growth (also single-cell growth) could be established for different carbon sources. In addition to the charged minimal medium we have also looked at cultivations with the commonly used BHI medium. Cultivations in BHI medium show the highest growth rate and the longest cells on average.

The dependence of cell morphology on the available nutrients has to be addressed in future experiments. Information on specific parts of the machinery that regulates cellular growth in bacteria [174, 175], and more specifically in *C. glutamicum* [38, 176, 177], has recently been published but does not relate to medium composition and cultivation conditions.

12 Modeling SOS and Phage Dynamics

This chapter is based on the publication “Live cell imaging of SOS and prophage dynamics in isogenic bacterial populations” [178].

DNA of viral origin is a prevalent element of bacterial genomes and can account for up to 20% of the whole genome [179, 180]. Genomically integrated bacteriophage DNA not only comprises fully functional prophages which are able to undergo a lytic life cycle, but also cryptic prophages or single phage genes which were trapped in the genome due to genomic rearrangements and gradual decay.

Due to the integration into the genetic circuitry of the bacterial host, these elements may have a significant impact on host fitness by equipping their host with genes for virulence factors or toxins [181–183], stress resistance [184] or metabolic traits [185]. Typically, the lysogenic, dormant state of temperate phages is very stable and maintained by action of a central phage repressor protein [45]. Early in the last century, however, free phage particles were found in cultures of lysogenic bacteria in the absence of an external trigger, leading to the term spontaneous prophage induction (SPI) [186, 187].

SPI was long considered as a potentially detrimental process, since a certain fraction of cells is continuously lost by the activation of lysogenic phages. Remarkably, several recent studies revealed the beneficial impact of SPI on the fitness of bacterial populations as it, for instance, contributes to the release of extracellular DNA (eDNA) which represents an important component of several microbial biofilms [188, 189], to the release of toxins or adhesion factors [190, 191], and has an important impact on horizontal gene transfer [187, 192, 193].

DNA damage, causing the induction of the cellular SOS response, represents the best-studied trigger for the switch of lysogenic phages to the lytic development [45]. As a result of DNA damage, the occurrence of single-stranded DNA (ssDNA) is sensed by the protein RecA, which binds to ssDNA, oligomerizes, and enters an active state, RecA*, in which it triggers the autoproteolysis of the SOS repressor protein LexA [194]. Derepression of the SOS genes results in the expression of more than 40 genes involved in DNA repair, recombination, and inhibition of cell division.

In previous studies, single-cell analysis of reporter strains revealed a spontaneous activation of the SOS response caused by the sporadic occurrence of DNA damage [195–197]. Previous fluorescence microscopy and flow cytometry based approaches were not able to unravel the transient character of this stress response. It is reasonable to assume that a prolonged induction of the SOS response by sporadic DNA damage will also impact the activation of lysogenic phages in a certain fraction of cells.

The genome of *C. glutamicum* strain ATCC 13032 contains four cryptic prophage elements (cf. Subsection 2.1.2). The largest prophage CGP3 (187 kbp) was shown to be induced in a small fraction of cells under standard cultivation conditions [198]. Flow cytometry based studies suggested spontaneous SOS induction as an important trigger of CGP3 SPI [46].

In this study, we have applied an MSCC setup with live-cell imaging to monitor the dynamics of the SOS response and prophage induction in single cells of *C. glutamicum* populations. This time-lapse image based single-cell analysis of reporter strains supported the hypothesis that the sporadic induction of the SOS response is an important trigger of CGP3 SPI, but also disclosed a considerable fraction of SOS-independent SPI.

12.1 Results

Previous studies disclosed the spontaneous induction of the CGP3 prophage in *C. glutamicum* populations and revealed the SOS response as a prominent trigger of CGP3 SPI [46, 198]. These flow cytometry based analyses of reporter strains, however, were not able to resolve the ongoing phenotypic dynamics.

Therefore, we set out to monitor SOS response and prophage induction in single bacterial cells at spatio-temporal resolution using time-lapse microscopy. For our study, we have constructed a *C. glutamicum* ATCC 13032 strain that contains a genomically integrated P_{recA}-*venus* fusion and a plasmid-encoded phage reporter where P_{lys} was fused to *crimson*.

Treatment with the antibiotic Mytomycin C (MMC) resulted in a significant induction of the SOS reporter showing normally distributed single-cell fluorescence intensities. In contrast, the output of the phage reporter P_{lys}-*crimson* showed two peaks of reporter fluorescence reflecting the bistable nature of the decision between the lysogenic and lytic state of CGP3 (Figure E.5).

12.1.1 Spontaneous Induction of SOS and Prophage CGP3

To enable time-lapse fluorescence microscopy under stable micro-environmental conditions, we performed growth experiments of *C. glutamicum* in microfluidic LOC devices (for a detailed description of the setup, see Section E.1). Single cells of the dual reporter

strain *C. glutamicum* ATCC 13032::P_{recA}-*venus*/pJC1-P_{lys}-*crimson* were seeded into the chambers and growth and fluorescence in standard glucose minimal medium (CGXII + $4\,\%$ glucose) was monitored for roughly $8\,\mathrm{h}$ (cf. Figure 12.1, Figure E.2). Lineage information (Figure 12.1B) and single cell fluorescence traces (Figure 12.1C) were extracted for 20 independent microcolonies (4248 cells) with our semi-automated image analysis tool.

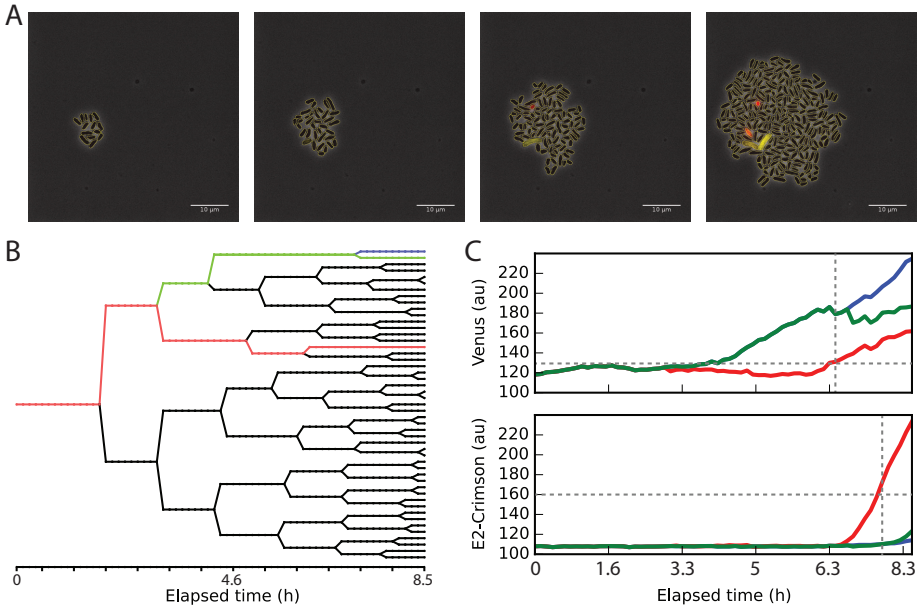


Figure 12.1: Live Cell Imaging of SOS and prophage induction. A. A series of processed images of a selected microcolony (MGC-16). Yellow contours denote the detected cell boundaries that were used for further processing. **B.** Lineage tree with highlighted traces of selected cellular events for image sequence of A. **C.** Fluorescence traces of Venus and Crimson reporters of traces highlighted in B. Dashed lines show time points where the reporter output passed the threshold (details on the computation in Figure E.3).

Under the chosen conditions, an average growth rate of $0.47\,\mathrm{h}^{-1}$ was observed which is in the range of the growth rate of *C. glutamicum* ATCC 13032::P_{recA}-*venus*/pJC1-P_{lys}-*crimson* determined in shaking flask cultivations. Using this experimental setup, $5.7\,\%$ of the cells exhibited spontaneous induction of the SOS response and $0.7\,\%$ displayed CGP3 SPI.

12.1.2 From Continuous Signal to Discrete Cellular State

We have defined eight discrete cellular states (Figure 12.2) based on the (continuous, thresholded) fluorescence reporter intensities of the dual reporter strain as well as the potential of an individual cell to generate offspring. The offspring state is determined by tracking a cell throughout an experiment to see whether it has daughter cells. Considering the fact that cells can only be in one of the states at one time-point at the experiment with the reconstructed lineage trees, enables us to investigate the fate of cells, i.e. the transition between states. We constructed a model consisting of cellular states and possible transitions, in which transition probabilities are extracted from experimental data.

The state of a cell is determined by applying an adapted threshold method (cf. Figure E.3) to each fluorescence reporter signal at a specific time point with thresholds $T_{SOS} = 130\,\text{a.u.}$ and $T_{Phage} = 160\,\text{a.u.}$ The threshold method also considers the history and future of cells to avoid over-interpretation of fluctuations in fluorescence intensities. That is, if the threshold is exceeded only marginally and for a short period of time (in hindsight) a cell will not be assigned the ON state (details in Section E.3). Furthermore, the thresholds have been set in a way that their influence on state transitions (in which we are ultimately interested in) is minimized (cf. Figure E.4). Additionally, the number of offspring for each cell is computed for determining the "Offspring" state of a cell.

Taking into account the temporal context of cells (predecessors as well as successors), the transition of cells between the states of two time points t_n and t_{n+1} (imaging interval, $\Delta t = 8$ min) was investigated. This is done by quantifying the number of state transitions in between two subsequent frames of acquired image sequences (Figure 12.2). Integration of experimental knowledge, i.e., image acquisition intervals (here $\Delta t = 8\,\text{min}$), enables us to compute conditional state transition probabilities (per Δt) from the previously obtained absolute counts.

Because probabilities are conditioned on different initial states, a direct comparison of transitions with different initial states is not advisable.

12.1.3 Transient Induction of the SOS Response

Previous studies have shown that the spontaneous induction of SOS-responsive promoters is likely due to the sporadic occurrence of DNA damage, including DNA double-strand breaks or stalled replication forks [46, 196, 199]. However, the snap-shot measurement of reporter output (e.g. using flow cytometry) obscures the transient nature of this DNA damage response. The frequency of state transitions from $\text{SOS}^+/\text{Phage}^-$ to $\text{SOS}^-/\text{Phage}^-$ (Figure 12.2) nicely illustrates the transient expression of the SOS reporter in a subset of SOS^+ cells. Whereas the majority of SOS^+ cells exhibits a prolonged induction, staying in an SOS^+ state for more than one imaging interval, $18.6\,\%$ of those cells recover from stress (re-entering the non-induced state and generating progeny). Whereas cells

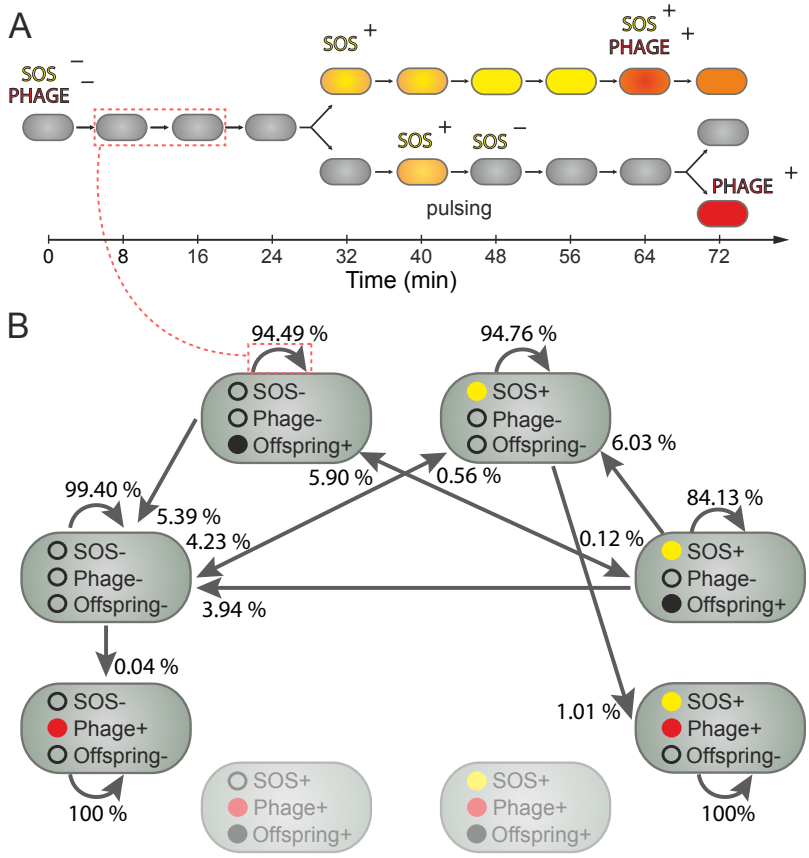


Figure 12.2: Frequency of state transitions. A. Illustration of the transitions between the four states based on the fluorescence reporter intensities ($SOS^-/Phage^-$, $SOS^+/Phage^-$, $SOS^-/Phage^+$, $SOS^+/Phage^+$). Fluorescence images of all cultivation chambers were taken every eight minutes and the states of all cells were compared to the states found in the previous image. The red box shows the matching transition in the Markov Model. **B.** Markov model of cellular states and transitions. Rounded rectangles denote states, while arrows denote possible transitions between the states. The arrow points from the initial state to the destination state. Values on the arrows show the conditional probability that a cell that was in the initial state at t_n is in the destination state at t_{n+1}. For details on data evaluation and the calculation of state transitions see Section E.1 and Figure E.3, respectively.

which recover from SOS induction show an overall reduced growth rate their growth rate is not significantly altered before induction in comparison to uninduced cells (cf. Figure E.1)).

The capability of cells to recover from SOS$^+$ is also reflected by the fact that more than one offspring originates from SOS$^+$ cells (Figure 12.3). This is particularly interesting considering that formerly SOS$^+$ cells have suffered from an increased mutation rate, but resume growth and thereby significantly contribute to the overall mutation rate of the whole population. On the other hand, prolonged SOS induction is followed by the induction of the CGP3 prophage (SOS$^+$/Phage$^+$) in a considerable fraction of SOS$^+$ cells ($10.1\,\%$ versus $0.7\,\%$ in the overall experiment).

12.1.4 CGP3 SPI Leads to Cell Death/Senescence

In contrast to the transient nature of the SOS induction, activation of the phage reporter coincides — in all cases — with a stop of cellular growth and likely cell death. This is in agreement with the cellular state and transition model (cf. Figure 12.2), where we do not observe transitions into one of the Phage$^+$/Offspring$^+$ states. Furthermore, the model topology (as extracted from the experimental data) shows that cells entering a Phage$^+$ state remain in this state throughout the experiment (cf. Figure 12.3 and Figure 12.4). Thus, this live cell imaging approach nicely demonstrates that induction of the cryptic prophage CGP3 results in cell death or senescence of the affected cells.

12.1.5 Correlation of SOS and prophage induction

Previous flow cytometry-based studies suggested that spontaneous SOS induction is a prominent trigger of CGP3 SPI [46]. In the present study, we analyzed the correlation of SOS and phage reporter output at spatio-temporal resolution in order to visualize how often these two responses correlate at the single-cell level.

To that end, we evaluated the temporal development of all single cell traces exhibiting SOS and/or prophage induction over the course of the experiment and analyzed their correlation (Figure 12.4). In agreement with previous findings these data revealed that a significant fraction of SOS$^+$ cells ($>8\,\%$) also exhibited prophage induction over the course of the experiment (cf. Figure 12.4B). Furthermore, SOS induction preceded CGP3 activation in $>60\,\%$ of Phage$^+$ cells, supporting a causal connection between these rare single cell events (Figure 12.4C). Remarkably, a considerable fraction of SOS-independent SPI ($>30\,\%$) was observed under the chosen experimental conditions. Thus, these data clearly indicate the existence of further factors, besides the SOS response, that are involved in the activation of the cryptic prophage CGP3.

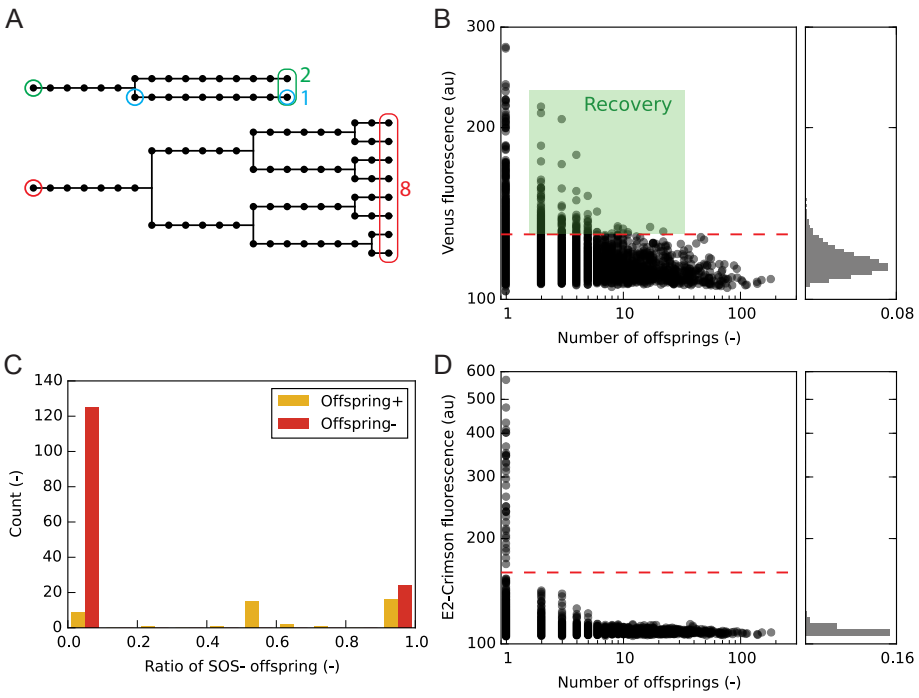


Figure 12.3: Recovery of SOS$^+$ and Phage$^+$ cells. A. Color-coded scheme for the computation of offspring of SOS$^+$ or Phage$^+$ cells. To compute the number of offspring of a cell c, the number of leaves of the subtree rooted at c is computed. Thus, cells that do not divide until the end of an experiment are denoted by an offspring of 1. **B.** Scatter plot of the number of offspring vs. the maximum Venus fluorescence (P_{recA}) of each generation. **C.** Normalized occurrences of ratios of SOS$^-$ offspring of SOS$^+$ cells that do not divide and spawn offspring, respectively. **D.** Maximum Crimson fluorescence (P_{lys}) per generation plotted against the number of daughter cells.

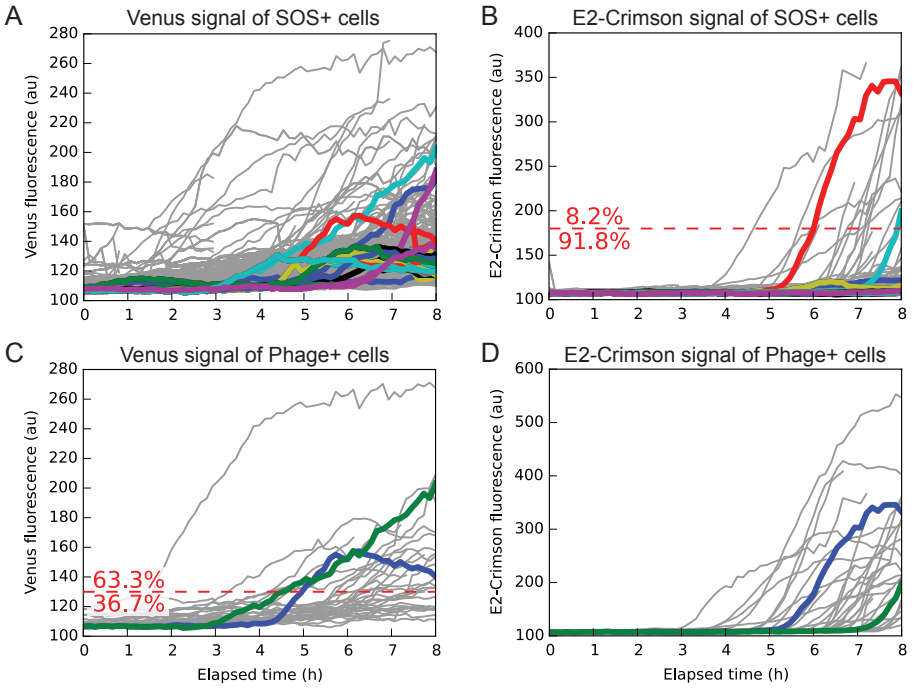


Figure 12.4: Correlation between spontaneous SOS and prophage induction. Colored traces are highlighted as examples from the complete set of traces (gray lines). **A.** Fluorescence traces (Venus) of all cells that have been identified as SOS^+ during the experiment. **B.** Fluorescence traces (Crimson) of all cells identified as $Phage^+$ during the experiment. **C.** SOS-dependent prophage induction: fluorescence traces (Crimson) of all SOS^+ cells. **D.** SOS-dependent and -independent SPI: fluorescence traces (Venus) of all $Phage^+$ cells.

12.1.6 Generation-Dependency of Spontaneous SOS and Prophage Induction

Recent studies suggested that the majority of sporadic SOS inductions occurs in a replication-dependent manner, e.g. due to polymerases stalled at replication forks [200]. In the following, we set out to compare SPI and SOS induction in exponentially growing versus non-growing cells.

In order to discriminate between time- and generation-dependent induction of SOS and prophage, respectively, we performed a starvation experiment, where the supplied medium was switched after $8\,\mathrm{h}$ of exponential growth to minimal medium without carbon source. After $\approx 24\,\mathrm{h}$ of starvation, the medium supply was switched back

to standard glucose minimal medium in which cells resumed growth. Whereas cells cultivated in the reference channels (+ carbon source) continued exponential growth, carbon-starved cells showed a significant drop in the growth rate, but exhibited residual growth for the next $\approx 12\,\mathrm{h}$ (Figure 12.5). After approximately $12\,\mathrm{h}$ of starvation, the total cell number reached a plateau and no significant increase in population area was observed for the following $\approx 10\,\mathrm{h}$. Remarkably, even in this time span spontaneous SOS induction occurred at a lower but still considerably high frequency ($1.35\,\mathrm{h}^{-1}$) suggesting the significant contribution of replication-independent mechanisms (cf. Figure 12.4). In contrast, no SPI was observed in the starvation phase. This is in line with the finding that prophage induction is only triggered in actively proliferating cells [201] (see also Figure E.6).

12.2 Discussion

In the present study, we used a state of the art microfluidics-based live-cell imaging platform to monitor SPI in *C. glutamicum* microcolonies with spatio-temporal resolution. These experiments provide detailed insights into the dynamics of prophage activation and stress responses at the single cell level and highlight the value of single-cell approaches in comparison to standard bulk assays.

The host SOS response represents the by-far best-characterized trigger of lambdoid prophages [194]. Interestingly, previous single-cell studies showed that a small fraction of cells exhibits a spontaneous induction of this stress response — likely caused by the sporadic occurrence of DNA damage [195–197]. Using a flow cytometry- and microscopy-based analysis of reporter strains, the spontaneous induction of SOS-responsive promoters (*E. coli*: *sulA*, *umuCD*, *lexA* and *recA*, *C. glutamicum*: *recA*) was measured to range between $0.09\,\%$ to $3.1\,\%$ [46, 196, 202]. These numbers reflect the high dependency of absolute values of suchlike measurements on the organism, the experimental procedures, and the data evaluation. More importantly, such values were generated to describe a stress response exhibiting a highly dynamic nature. A snap-shot measurement (e.g. flow cytometry analysis) of single cells at a certain time cannot reveal the actual fate, i.e., the development of particular cells over time. In order to visualize these transient changes of the SOS response, we applied a threshold ($T_{SOS} \approx 1.18 -$ fold mean), which resulted in a comparably high fraction of SOS$^+$ cells in this live cell imaging study ($5.7\,\%$). The setting of this threshold is also supported by the fact that SOS$^+$ cells displayed in almost all cases a significant drop in growth and showed an elongated morphology typical for an induction of SOS genes encoding cell division inhibitor proteins [203, 204].

Sporadic DNA damage, e.g., the spontaneous occurrence of DNA double-strand breaks, may for instance occur by the collapse of replication forks [200] and is a frequent cause for genomic rearrangements or mutation hot spots [205–207]. Under stress conditions this may significantly accelerate evolution of bacterial strains or communities [208,

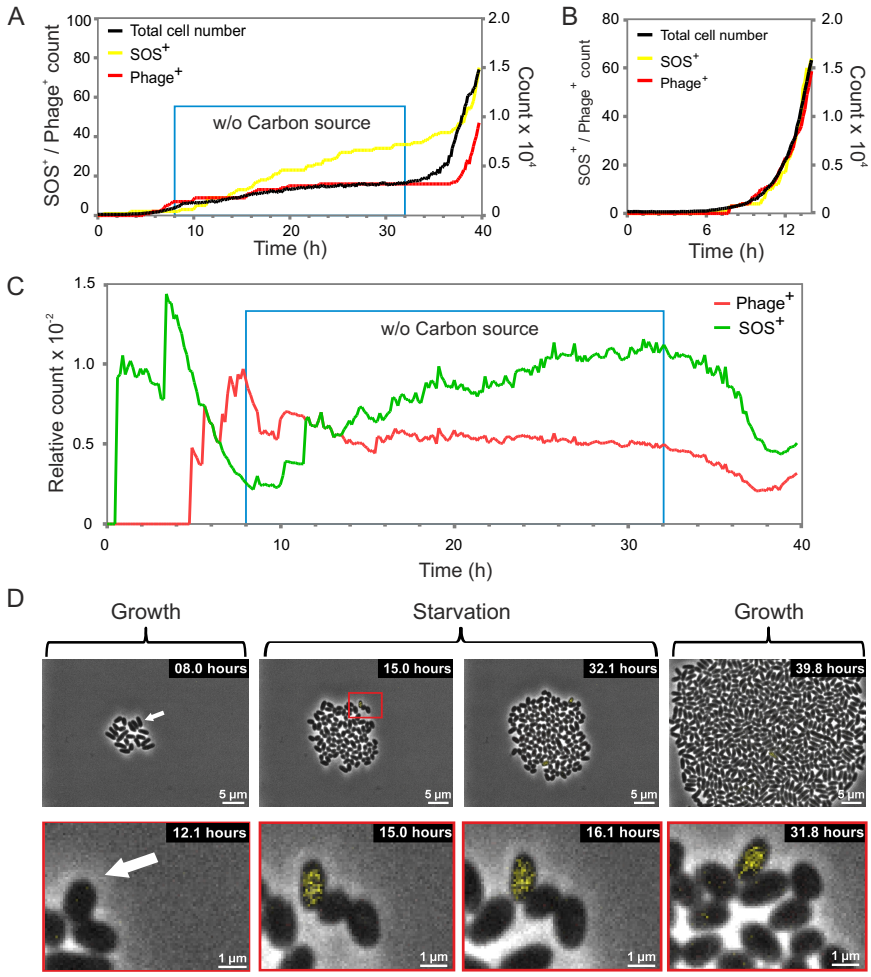


Figure 12.5: Time-dependent SOS and prophage induction in cells of the dual reporter strain ATCC 13032::P_{recA}-*venus*/pJC1-P_{lys}-*crimson*. **A.** Counts of the total cell number, SOS^+ and $Phage^+$ cells were plotted against the time. The starvation phase is shaded in blue. **B.** Control experiment where the cells were grown in the presence of a carbon source in standard CGXII minimal medium. **C.** The number of SOS and phage positive cells divided by the total cell number plotted against the time. During the starvation phase, SOS induction is still increasing whereas no additional SPI events were observed. **D.** Picture series of a selected microcolony cultivated as described in **A**. One cell exhibiting SOS induction during the starvation phase is highlighted (white arrow).

209]. In contrast to previous studies that have measured spontaneous SOS induction, the live-cell imaging approach of this work allowed not only the quantification of the frequency of spontaneous induction but also enabled to measure the fraction of cells ($18.6\,\%$) that recovered from this stress. In the course of SOS induction, affected cells have likely suffered from an increased mutation rate, e.g. due to the action of error-prone polymerases. Consequently, recovering cells likely have considerable impact on the overall mutation rate of the population; thereby also affecting its adaptability and potential to evolve [210]. A recovery of about $18.6\,\%$ is in a similar range as previous measurement of *E. coli* SOS$^+$ cells, where about $35\,\%$ of the cells isolated via FACS formed colonies on plates [196]. The authors concluded from negative propidium iodide (PI) staining that the rest of the SOS$^+$ cells remained in a senescence-like, dormant state. However, previous studies ignored another factor impacted by a spontaneous induction of the SOS response, namely the activation of lambdoid prophages, mobile elements or cryptic (degenerated) prophages. In contrast to SOS, their induction often leads to a "dead end".

In this work, we could show that under the chosen assay conditions more than $8\,\%$ of the SOS$^+$ cells also induced the cryptic prophage CGP3 (Figure 12.4). In all cases, the signal of the SOS reporter preceded the signal of the phage reporter supporting the causal connection between the SOS induction and prophage induction. Due to the limited observation time, this value, however, rather represents a lower bound for SOS-induced CGP3 induction.

Remarkably, this study could also demonstrate a considerable high fraction of SOS-independent CGP3 induction ($>30\,\%$). This is in line with the finding that a ΔrecA mutant still exhibits CGP3 induction in $0.12\,\%$ of the cells. Alternative (RecA-independent) pathways for the induction of lambdoid prophages have been described in a few studies and include for instance the induction of *Pseudomonas aeruginosa* prophages by the accumulation of acyl-homoserine lactones (AHL) as a density-dependent mechanism [211] or the RcsA- or DsrA-mediated induction described for *E. coli* [212]. The identification of SOS-independent induction mechanisms of CGP3 will be target of future studies.

DNA replication has been previously reported as an important source for the occurrence of sporadic DNA damage, e.g. replication fork collapses [200]. Recent single-cell studies also have addressed the question whether spontaneous SOS induction is triggered by a generation- or time-dependent mechanism [196, 197]. Those studies suggested that spontaneous DNA breakage is precisely correlated with the number of cell divisions [197]. In contrast, the starvation experiment conducted in this study clearly revealed continued SOS induction in the starvation phase (cf. Figure 12.5). Given that colony area and cell number stayed constant for about 10 hours these data suggest that also time-dependent mechanisms feed into the spontaneous induction of the SOS response. In fact, it is possible that, for instance, oxidative stress or changes in iron availability may cause spontaneous DNA damage in resting cells [213]. In contrast, CGP3 induction

occurred only in proliferating cells which is in agreement with the finding of Pearl *et al.* who reported that persistent cells, existing in a dormant state, are protected from prophage induction [201]. In fact, persistence might represent an important mechanism of bacterial populations and communities to avoid complete eradication upon prophage induction under certain stressful conditions.

In all cases, SOS-dependent and SOS-independent, induction of the cryptic prophage CGP3 led to a stop of growth and likely cell death. However, lysis of the affected cells was only observed for a few examples, which could, however, also be an artifact of the experimental approach since secreted hydrolytic phage enzymes might be removed by the continuous medium flow applied during microfluidic cultivation. During microscopy on agar pads, CGP3 induction frequently coincided with the lysis of the particular cell [198]. If CGP3 induction, or, more generally, the induction of several (cryptic) prophages, leads to cell death, why does this degenerated viral DNA remain in bacterial genomes in the course of evolution? Which selection pressure enforces the maintenance of degenerated prophage elements? Interestingly, several recent studies highlighted the beneficial effects prophages may have on the fitness of their bacterial host (reviewed in [187, 214]. Cryptic *E. coli* phages were reported to contribute significantly to the overall fitness and stress resistance of populations [184, 215]. Sacrificing a small fraction of cells by SPI is exploited by several bacterial strains for the development of biofilms [188, 189, 215–217], the release of toxins [190], or host microbe interactions [191, 218].

In conclusion, SPI represents a common but often unnoted phenomenon of lysogenic bacterial populations. This study provided a first insight into SPI at spatio-temporal resolution. Whereas our data confirmed the spontaneous induction of the SOS response as a prominent trigger evidence was provided for further RecA-independent factors contributing to SPI in bacterial populations. Future studies will reveal how host-phage interaction has shaped the diversity of trigger inputs to adjust SPI to an optimal level depending on the particular environmental and physiological conditions of the host.

12.3 Population Modeling

We have presented a model that cellular state and transition model with SOS^-/SOS^+, $Phage^-/Phage^+$, and $Offspring^-/Offspring^+$ states (cf. Figure 12.2). This model considers individual cells and connected states as entities. While this is useful for extracting and modeling information on single-cell level, it does not take into account effects that might occur on population level. A prominent example for population effects is the emergence of persister cells (resistance to antibiotics) [219, 220] as well as bet hedging strategies [221]. Such bet-hedging strategies offer a species an evolutionary advantage in fluctuating and unpredictable environments to which the species has to adapt.

It is a longstanding question how such population behavior is implemented and how it evolves. Is it the behavior of individual cells that leads to a consistent and interpretable

population behavior? To investigate the reason, we have adapted the cellular state and transition model shown in Figure 12.2 to a stochastic population model. This model is initialized with a number of cells in $SOS^-/Phage^-$ state that change states (via possible transitions) over time. We have modeled the $Offspring^-/Offspring^+$ states implicitly by allowing cells from $Phage^-$ states to generate offspring of the same state. The offspring will be subject to the same transitions as other cells.

The model has subsequently been implemented in StockKit2 [222]. The complete set of equations is provided in Table E.1. The results of an exemplary simulation of the implemented model is shown in Figure 12.6. In this simulation, we have simulated an external stress that effectively increases the parameter for state changes to SOS^+ states while the stress is applied. The duration of the stress signal is increased during an experiment to see the influence on the complete population. Once the simulated stress pulse has vanished, the population recovers, if cells in an $Phage^-$ state have survived. The observation seems reasonable since cells are able to recover from SOS^+ states. Nonetheless, there is a high dependency between the outcome of the simulation and the parameters for state transitions. In this particular case, the transitions from $SOS^+/Phage^-$ to $SOS^-/Phage^-$ and $SOS^+/Phage^+$ have the same initial state. Hence, the ratio between the transition probabilities to the destination states influences the outcome of the population simulation.

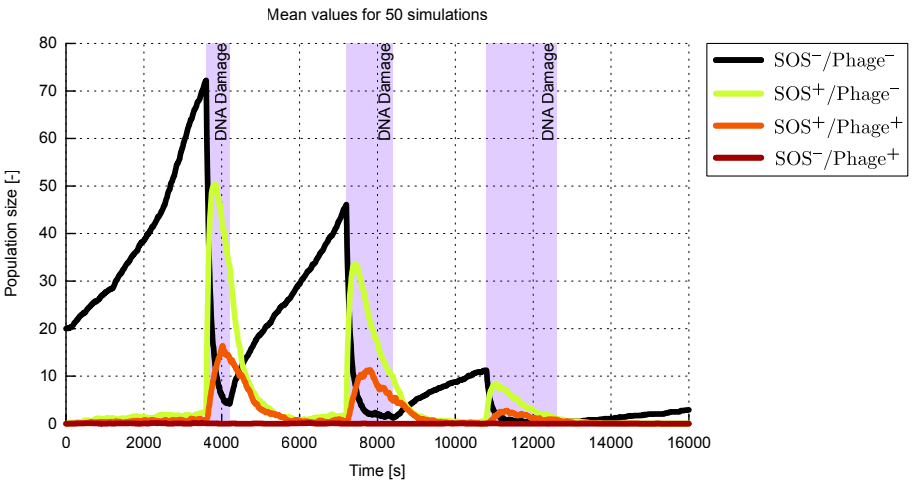


Figure 12.6: Simulation of the population model with three perturbations of different length (increasing). The reaction of the population and the regeneration after each pulse can be observed.

12.4 Approaching Molecular Modeling of SOS and Phage Dynamics

While the previously presented models are purely based on observations from SCA, the interesting question of how the observed patterns are generated on a molecular level remains unanswered. A fact that is hampering investigations, is the general lack of knowledge about molecular mechanisms in *C. glutamicum*. Reports on the structure of the bacterial SOS network have identified key genes and proteins, however, mostly in the model organism *E. coli* [223–225]. It was reported that also in *C. glutamicum* the gene regulatory network of the SOS response has a complex structure involving the LexA and RecA proteins [226]. Genes under control of LexA code for proteins with a variety of physiological functions, but for nearly half of them their function remains unknown [227].

To model the connection between the SOS response in *C. glutamicum* and the prophage CGP3, we have adapted the phage λ and an SOS response model from *E. coli* [228] to *C. glutamicum*. The two systems, i.e., SOS and prophage induction, are connected via a yet to be identified key protein or regulatory pathway, denoted as X in Figure 12.7.

While key players of the SOS response have been extracted from literature, crucial genes and proteins for the prophage induction have been experimentally identified using transcriptome analysis of stressed *C. glutamicum* populations. Because the SOS response triggers prophage induction in *C. glutamicum*, prophage regions are actively transcribed in stressed cells [46]. The most robustly expressed genes have been selected as reporters for prophage induction and are, hence, also essential parts of the constructed molecular model. Results of the transcriptome analysis are shown in Figure 12.8.

12.4.1 Realization of the Model

Complex regulatory processes are involved in the coupling of the SOS response with the CGP3 induction in a single model. This is mostly owing to the fact that key proteins, e.g., LexA, are acting as transcriptional regulators for other genes, e.g., *recA*. Therefore, the model has to incorporate the complete process from transcription, translation, to regulation, for each involved player. Since SOS response as well as the prophage induction are dynamic (and stochastic) process at low molecule concentrations in a cell, processes have to be modeled in a stochastic framework. Biological knowledge is incorporated into this framework via species (e.g., promoters, mRNA, proteins) and reactions that are transforming species. In a stochastic framework, reactions are not running at fixed rates but are executed with a given probability. This is of utmost importance for systems that show undeterministic behavior because of low concentrations of proteins (up to hundreds of molecules). Because we are looking at a small number of promoters, stochastic effects might drive the transcription of key regulators.

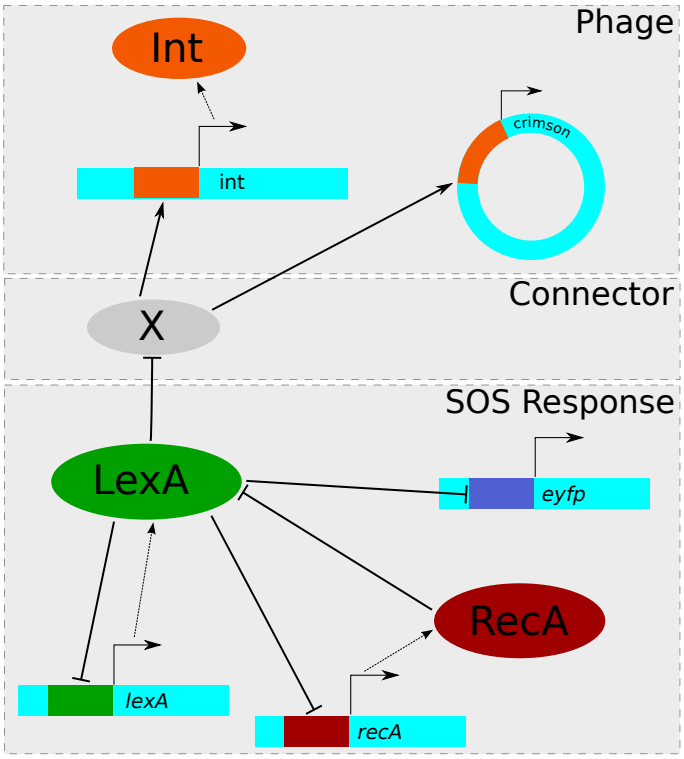


Figure 12.7: Illustration of the regulatory processes and connections between the SOS response and the prophage induction. LexA and RecA are well known components of the bacterial SOS system. Int is a putative integrase that is responsible for the excision of the prophage (if activated). eYFP and Crimson are used as fluorescence reporters for the SOS response and the phage induction, respectively. X denotes the yet to be determined connector between the two systems.

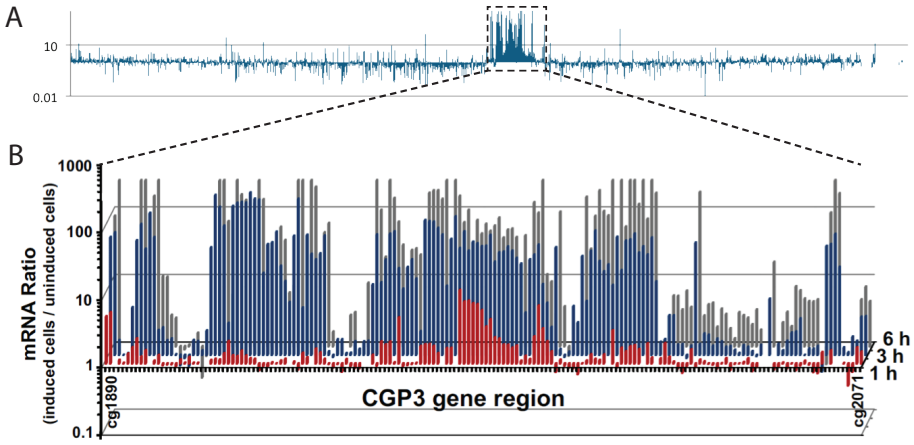


Figure 12.8: Identification of phage promoters. Time-resolved transcriptome analysis of stressed *C. glutamicum* cells upon treatment with antibiotics (Mitomycin C). **A.** Fold increase of transcription before and after treatment with Mytomycin C. The zoom window denotes the prophage CGP3 region. **B.** Fold increase of transcription in the CGP3 region at $1\,\mathrm{h}$, $3\,\mathrm{h}$ and $6\,\mathrm{h}$ after induction (courtesy of Frunzke lab).

Since transcription, translation, and regulation reactions are similar for different proteins, one example is picked and described in detail with matching equations.

Example: RecA

The creation of one protein (RecA, LexA, EYFP) comprises several steps:

Regulation: Free promoter is bound by a regulator

Transcription: DNA is transcribed to matching mRNA if a free promoter is available

Translation: mRNA is translated into protein

The transcription of *recA* is negatively regulated by the amount of LexA molecules in a cell. The transcription is modeled as $f_r \longrightarrow f_r + m_r$, where f_r denotes a free promoter of *recA* and m_r a *recA* mRNA molecule. This reaction does not directly incorporate the regulation of the *recA* promoter (P_{recA}) by LexA molecules. However, the regulation is achieved by transiently converting free promoters f_r to the bound state b_r by means of a LexA molecule: $l + f_r \longrightarrow b_r$. Promoters in a bound state cannot generate mRNA molecules. The complex of LexA and a promoter is of transient nature and LexA can dissociate to free the promoter ($b_l \longrightarrow f_l + l$). Furthermore, mRNA molecules undergo degradation: $m_r \longrightarrow \emptyset$.

In a second step, mRNA is translated into protein: $m_r \longrightarrow m_r + r$. Additionally, RecA marks LexA for cleavage ($r + l \longrightarrow r$), effectively removing LexA molecules from the system.

While the previous description of RecA acts as an example, this scheme can be directly translated to the other promoters, mRNAs, and proteins of the model. The complete set of reactions and parameters used for the simulation studies are provided in Table E.2 in Section E.5. All in all, the model contains about 40 parameters for reactions and the initial molecule numbers as additional inputs. Initial values as well as parameter values for the SOS network have been adapted from [224].

Simulations with the proposed model (Figure 12.9) reveal that the regulation scheme is able to model the recovery of a cell from external stress with the phage integrase remaining active. All in all, the proposed model captures the characteristics of experiments with a set of adapted parameter values.

The parameter values that are underlying the model, have either been extracted from literature or are based on educated guesses. Additionally, the model simplifies complex biological processes into single reactions (e.g., translation) to which a single parameter is assigned. Therefore, most of the parameters of the model cannot be biologically interpreted and can hence not be determined experimentally. An alternative to the direct measurement of parameters are forward simulations that are fitted to single-cell data from microfluidics using eYFP and Crimson signals. The use of forward simulations of stochastic models for parameter estimations is a current topic of research where too little knowledge is yet available for complex applications [229].

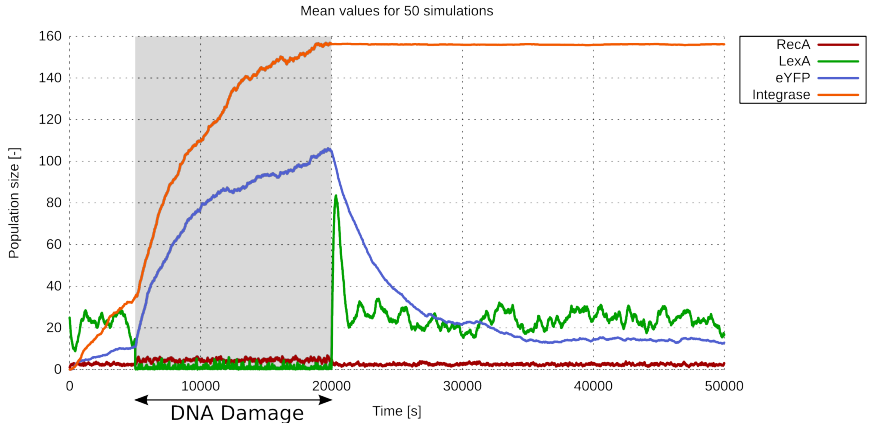


Figure 12.9: Stochastic simulations of the presented gene regulation model. Overall simulation time about $72{,}000\,\mathrm{s}$. An antibiotics pulse was simulated from $500\,\mathrm{s}$ to $2000\,\mathrm{s}$. Afterwards, the system returns to the initial state. Plot shows the mean values for 50 stochastic simulation of the described system.

Part IV

Summary and Outlook

13 Summary

Microfluidics-based live-cell imaging shows a great potential for investigating bacterial processes on a per cell basis. The full potential, however, can only be harnessed when using image analysis approaches for the extraction of time-resolved, quantitative information from image sequences.

We have described the challenges that are imposed by the experimental setup for an image analysis pipeline in the introduction of Part II. Driven by these challenges, a customized image analysis pipeline for the investigation of *C. glutamicum* in LOC devices has been developed and implemented as a ready-to-use software package (cf. Chapter 6) that is applied by experimentalists. Advanced visualization techniques are required to make the plethora of data easily accessible. To facilitate investigations on single-cell data we have developed the visualization and analysis software *Vizardous* (cf. Chapter 8).

Several applications have shown the quality of data that is extracted from acquired image sequences. The investigation of cellular heterogeneity in populations of *C. glutamicum* is enabled by the combination of methods from different fields (engineering, biology, computer science). Especially, investigations of microbial growth with respect to nutrients and medium composition is of great interest for bioprocess optimization. We have shown, that the implemented image analysis pipeline has been used for the characterization of microfluidic platforms (Chapter 9). Furthermore, LOC devices are used for screening purposes in the context of cultivation optimization when combined with advanced imaging and image analysis techniques (Chapter 11).

In addition to questions of microbial growth, the heterogeneity in the activation of the prophage CGP3 in prominent *C. glutamicum* strains has been investigated in depth (cf. Chapter 12). Experimental results did not yield further insights into the molecular underpinnings of the trigger. We were, however, able to show that the induction of CGP3 might be a strategy to kill cells that have experienced DNA damage. This could be seen as a means to remove those cells from a population although they might recover from the damage.

Section 12.4 has presented first principles of a gene regulatory network that combines the microbial SOS response with the induction of the prophage CGP3. Pushing the modeling of the molecular regulation of the CGP3 induction further requires further data from transcriptome and protein interaction analysis to identify further players in the activation process and set up hypotheses. In addition to refining the topology of the gene regulatory network, model parameter have to be determined experimentally or with computational means (parameter estimations for gene regulatory networks).

The applications of the presented image analysis pipeline that have been described in this work (cf. Part III) are only an excerpt of the whole range of possible studies. The pipeline, including *Vizardous*, has been used in a multitude of studies by researchers from different [26, 46, 140, 141]. The JuNGLE suite for image analysis in conjunction with *Vizardous* has been established as the prevalent solution for SCA of rod-shaped bacteria at IBG-1.

14 Outlook

The presented image analysis pipeline is used for the extraction of time-resolved, quantitative data, therewith contributing to or enabling different applications. Still there is room for improving different modules by incorporating additional knowledge that is available but also continuously generated in experiments. Moreover, a sophisticated way of storing information and making them more accessible is desired. This chapter will highlight several projects that are worth further investigation.

14.1 Beads for Comparable Fluorescence Intensities

To show biological reproducibility, biological experiments are in most cases executed in at least triplicates. This approach is furthermore used to effectively detect outliers in experiments. This line of action, however, proves to be a challenge for experiments on microfluidic chips. First of all, chips are manufactured manually and, hence, show production variability. As second challenge are minor deviations in experimental procedures between experimentalists. However, the biggest technical challenge is fluorescence microscopy with respect to comparability. Time-resolved fluorescence intensity signals heavily depend on the configuration (i.e., exposure time, used light source) and adjustment of the microscope. Hence, a method for the quantitative comparison of experiments that have been not been executed in parallel (with same settings on the same microscope) is desirable.

To that end, a normalization element has to be added to experiments which shows constant fluorescence signals between experiments. One way to tackle that challenge is the introduction of fluorescent (polystyrene) beads into the system. Using the signal of beads, that by manufacturing have the same amount of fluorescent protein enclosed, will result in normalized fluorescence intensities for the biological specimen. This, however, requires a tightly controlled manufacturing process and a thorough characterization of the beads.

A first approach would be to introduce the beads and image them before the seeding phase has even been initiated. Later on, all extracted fluorescence values are normalized

to the mean value of selected beads. The goal is to generate a reference frame for relative fluorescence intensities instead of absolute ones that are generally regarded as less robust.

The presented image analysis pipeline can be used to detect beads in an image and track them over time. While this adds additional computational burden, it enables the adaptation to changes in the experimental setup after the experiment has already been executed. Nonetheless, establishing an normalization step as a postprocessing measure is only the first step. Using the generalized data structures of imglib2 and ImageJ2 it is possible to implement normalized intensity images in such a way, that implementations working with images do not have to be adapted. Only the intensities that are returned would be normalized on the fly. To achieve this objective, a normalization matrix could be extracted automatically and imported for a dataset that should be normalized according to this matrix.

14.2 Improve Detection of Regions of Interest

The presented approach for the detection of regions of interest (see Section 6.4) operates on complete images. Although the LOCs are designed in-house and schematics are readily available, this additional information is currently not integrated. Additionally, the chip designs feature a cross-like structure that has more prominent features than the MGC as shown in Figure 14.1. These cross-like structures cannot be overgrown by the bacterial population during a cultivation and are, hence, detectable throughout an experiment. Once, this cross has been detected, it is easy to find the matching region of interest for an alignment cross based on the MGC design.

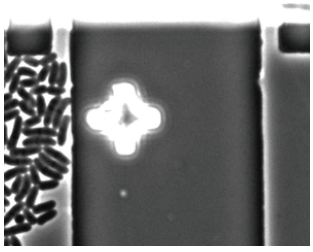


Figure 14.1: Alignment cross on LOC device located between two MGCs.

Additionally, this alignment cross can be used for tackling the registrations problem. As described in Section 3.1, the registration procedure operates in a hierarchical way on the complete image. Due to the runtime of $\mathcal{O}(n^2)$, quartering the size of the images to be registered will decrease the computational demand of the computations by a factor of 16 for each image of a sequence. This reduction could be facilitated because the alignment cross' registration parameters can be applied to the complete image.

14.3 Incorporating Additional Knowledge into the Tracking Process

Also, there is room for improving different modules by incorporating additional knowledge that is continuously generated in experiments. Especially the tracking module could benefit from the integration of additional knowledge. This knowledge is available in different forms like integrating the approximate duration of one cell cycle to render splits at the end of a cell's lifetime more likely than otherwise.

Another improvement that is worth exploring in future is the integration of population information for tracking individual cells. While this may seem counterintuitive, the assumption that cells are moving independent of each other in a population does not hold for our experimental setup. In the MGCs, a population of cells inhabits a very restricted volume. Hence, the per cell increase in volume will translate (since the height is restricted) into a spatial movement. Therefore, tracking results can be significantly improved by taking into account the population structure and an extrapolated growth direction in dependence of the position of cell in the population.

Approaches that integrate a priori knowledge have been investigated by Axel Theorell. He has worked on a Bayesian approach for reconstructing lineage trees from time-lapse image sequences. In the so-called multiple hypothesis tracking (MHT) approach [230, 231], possible reconstructions (q^k) are constructed in a framework based on Bayes' theorem. Here, a distribution of possible reconstructions is computed from observed cells denoted as Z^k. In an iterative procedure, reconstructions for a frame k are generated from lineage trees up to the previous frame $k-1$:

$$p(q_k|Z^{k-1}, q^{k-1}) = p(\Psi_k|Z^{k-1}, q^{k-1})p(q_k|Z^{k-1}, q^{k-1}, \Psi_k) \tag{14.1}$$

This approach allows first for the quantification of uncertainty in lineage tree reconstructions and, second, for the integration of additional knowledge in form of a priori probability densities.

14.4 Centralized Computation

The focus of the implemented image analysis workflow was on usability and applicability. Nonetheless, a future goal is to further robustify and improve the automatic execution of modules that currently require user interaction. It is in practice possible to run Fiji with the JuNGLE plugin on a terminal server that currently has to provide a user interface for interaction with JuNGLE. In contrast to an interactive mode that requires user interaction, most modules of the image analysis pipeline can be initialized and then executed in a batch mode.

The already established OMERO server (cf. Section 2.4) can for instance act as a central node for the distribution of computation task. The idea is, to have users upload

their experimental data into OMERO and assign it a processing template with defined parameters. The processing template will be executed in a headless mode, and the extracted information is again stored in OMERO. After a computation job is finished, the user is informed via e-mail and is able download or view the resulting data in an OMERO client.

First functionality is already in place to support this long-term goal, namely the imagej-omero [232] project that lets users execute ImageJ plugins (more specifically: wrappers for ImageJ operations) on an OMERO server. The surrounding infrastructure, i.e., distributing task, storing results in OMERO, and notifying users, has yet to be established.

In future, this distributed analysis structure can potentially improve data analysis and modeling tasks by evaluating more experiments.

Part V

Appendices

A Image Analysis

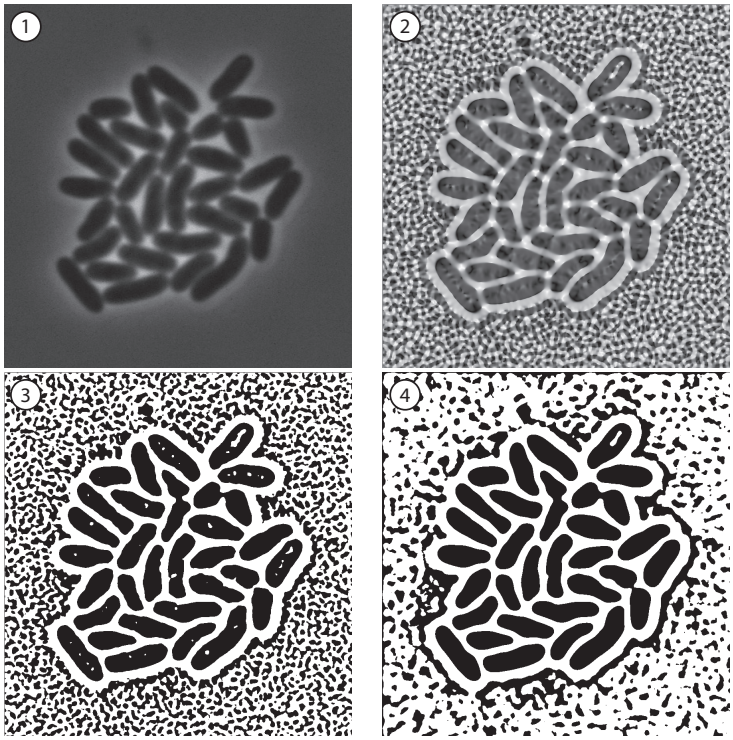


Figure A.1: Postprocessing of the Shape Index Map. **1.** Input image of a *C. glutamicum* population, I. **2.** SIM of the input image, denoted I_{sim}. **3.** Thresholded SIM, $I_{thresholded\_sim}$. **4.** Median filter version of $I_{thresholded\_sim}$, denoted $I_{cleaned\_sim}$.

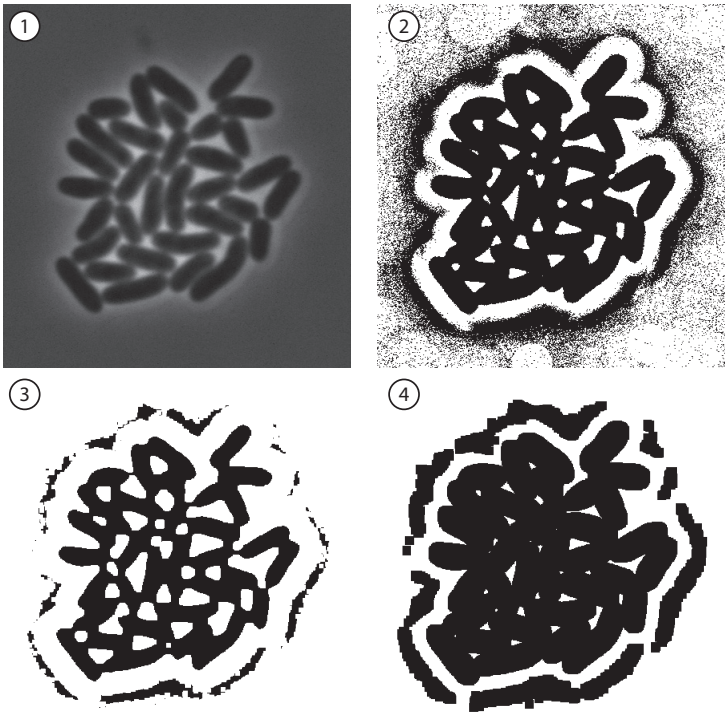


Figure A.2: Processing of images for carving out the microbial population. **1.** Input image I of a *C. glutamicum* population. **2.** Locally thresholded version I_{local} of I. **3.** Several erosion operations applied to I_{local} for removing speckles, denoted as I_{eroded}. **4.** Several dilation operations applied to 3, resulting in I_{mask}.

A.1 JuNGLE Features

- *MasterPlugin* for user interaction (i.e., configuration of pipeline parameters) and control of the overall image analysis pipeline (Section 6.2)
- *PreprocessorPlugin* registers image sequences and detects growth sites in images (Section 6.2)
- *SegmentationPlugin* detects cells in images and generates and an overlay of ROIs (Section 6.2)
- *Filters* allow for the filtering of detected cells for further processing (Section 4.5)
- *BottleneckDetector* separates touching cells based on knowledge about cell morphology (more information in Section 4.5)

- *SnakesPlugin* implements a post-processing procedure for cell detection where *SegmentationPlugin* and *BottleneckDetector* have failed (Subsection 4.7.1)
- *OverlayDetector* converts ROIs to TrackMate data structures
- *JuNGLE Displayer* displays TrackMate data structures and allows for the correction of segmentation results after the conversion step with *OverlayDetector*

A.1.1 Tools / Macros

- *Splitting Tool*
  - Easy selection of cells (i.e., ROIs) for processing
  - Splitting cells with virtual split line for preview of result
  - Merge cells together
- *Generate Results Table Tool* gathers per-frame information of microbial populations
  - cell number
  - cumulative cell area
  - cell area mean
  - cell area standard deviation
  - fluorescence intensity mean
  - fluorescence intensity standard deviation
- *Generate Traces Tool* for bead characterization
  - Takes ROIs in the first frame and computes fluorescence intensity traces for the whole image sequence
  - Output as Results Table that can conveniently be exported to CSV and XLS(X)
- *Add ROI to Overlay Tool* takes a ROI (which has been created with the built-in ImageJ selection tools) and adds it as cell to the cell detection overlay of an image

B Visualization

B.1 *Vizardous* Features

- Import trees from standard formats (PhyloXML)
  - from file system
  - from external source (OMERO)
- Visualize single experiments in lineage tree
- Visualize multiple experiments in lineage trees
- Annotate tree entities with extended information
- Analysis
  - of single-cell data
  - of all cells in an individual frames
  - of complete populations
- Visual comparison of cells from a single tree
- Visual comparison of multiple trees
- Filter tree elements according to structural information
- Filter tree elements according to meta information
- Sort trees according to meta/structural information
- Export trees
  - to vector graphics
  - to bitmap

- Export charts
  - to vector graphics
  - to bitmap
- Export tabular data

B.2 Libraries

Vizardous uses available (open source) libraries wherever appropriate. We have compiled a list of the external libraries that are used for building *Vizardous* in Table B.1.

B.3 MetaXML Description

This section provides a detailed overview of the information contained in a MetaXML file. An example file is provided in Listing B.1 for reference.

Listing B.1: Exemplary MetaXML file with a single cell in a one frame sequence.

```
<?xml version="1.0" encoding="UTF-8"?>
<metaInformation xmlns="http://13cflux.net/static/schemas/metaXML/2"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://13cflux.net/static/schemas/metaXML/2 metaXML-2.6.0.xsd">
  <projectName>default</projectName>
  <experimentDuration unit="min">512</experimentDuration>
  <frame id="0">
    <elapsedTime unit="min">0</elapsedTime>
    <population id="0">
      <center>
        <x unit="um">16.38</x>
        <y unit="um">29.53</y>
      </center>
    </population>
    <cell id="0">
      <center>
        <x unit="um">16.38</x>
        <y unit="um">29.53</y>
      </center>
      <length unit="um">2.52</length>
      <area unit="um^2">2.11</area>
      <fluorescences>
        <fluorescence channel="yfp">
          <mean unit="au">113.30</mean>
          <stddev unit="au">0.0144</stddev>
        </fluorescence>
        <fluorescence channel="crimson">
          <mean unit="au">106.95</mean>
          <stddev unit="au">0.0084</stddev>
        </fluorescence>
      </fluorescences>
    </cell>
  </frame>
</metaInformation>
```

Table B.1: External libraries used for building *Vizardous*.

Name	Version	URL
Java Universal Network/Graph Framework (JUNG)	2.0.1	http://jung.sourceforge.net/
JGraphX	3.1.2.1	https://github.com/jgraph/jgraphx/
JFreeChart	1.0.15	http://www.jfree.org/jfreechart/
JCommon	1.0.17	http://www.jfree.org/jcommon/
Apache Batik	1.7	http://xmlgraphics.apache.org/batik/
FreeHEP VectorGraphics	2.3	http://java.freehep.org/vectorgraphics/
iText	5.0.6	http://itextpdf.com/
Jaxen	1.1.6	http://jaxen.codehaus.org/
JDOM	2.0.5	http://www.jdom.org/
Java Excel API	2.6.12	http://jexcelapi.sourceforge.net/
opencsv	2.3	http://opencsv.sourceforge.net/
Simple Logging Facade for Java (SLF4J)	1.7.7	http://www.slf4j.org/
OMERO Client	5.0.0	http://www.openmicroscopy.org/
Apache Commons Math	3.3	http://commons.apache.org/math/
Apache Commons Lang	3.3.1	http://commons.apache.org/lang/

Both elements <projectName> as well as <experimentDuration> describe the general experiment. The <metaInformation> tag furthermore contains as many <frame> elements as there are in the image sequence from which the file pair is generated. Each <frame> is uniquely identifiable and can be connected to an image file as well as a time-point in the experiment (<elapsedTime>).

One frame contains information about the background fluorescence in this frame (which could be used for normalization purposes in future versions), as well as about the microbial population (<population>) that is present in this frame.

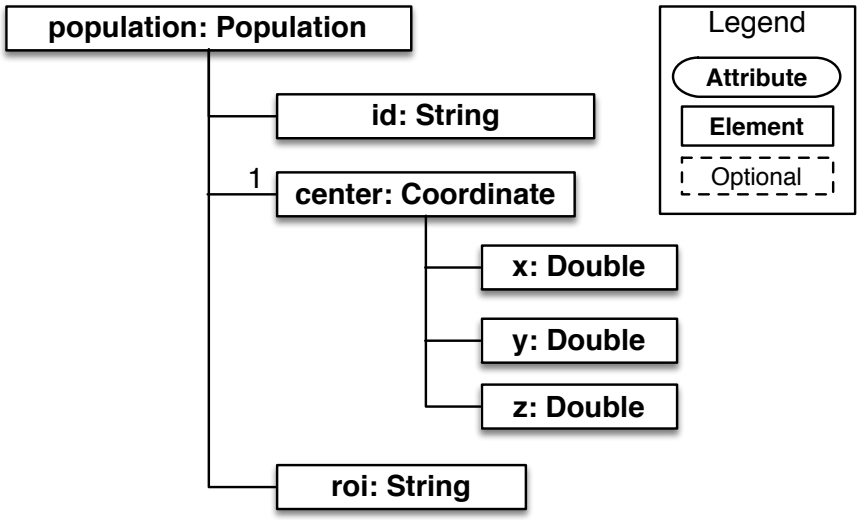


Figure B.1: Overview of the contents of the <population> tag.

Depending on the definition of the population, different characteristics are of interest. The information of a population mostly concerns the spatial (and via the connection to a specific frame also temporal) location of the point.

The <cell> element is, however, the most important part of the MetaXML definition because it describes the extracted characteristics of cells. Figure 8.2 provides a visual overview of the characteristics defined in MetaXML. Further information about the structure of the <cell> and <population> tag are provided in Figure B.2 and Figure B.1, respectively.

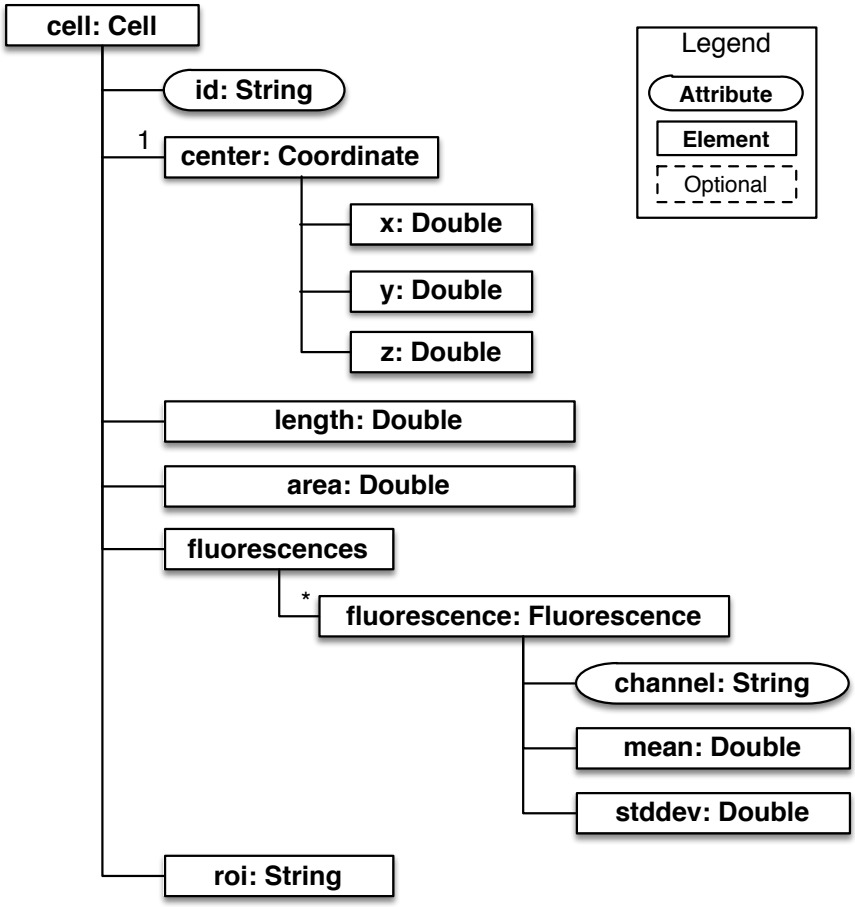


Figure B.2: Overview of the contents of the `<cell>` tag.

C Growth Quantification

C.1 Microfluidic Device Cultivation

An in-house developed microfluidic platform was used for *C. glutamicum* single-cell analysis [2, 59]. The microfluidic device incorporates a few hundred cultivation chambers with dimensions to ensure monolayer growth of isogenic microcolonies, with up to a few hundred cells maximum each. Phase contrast and fluorescence time-lapse imaging was performed at $15\,\mathrm{min}$ intervals.

Medium was supplied continuously to ensure stable and constant environmental conditions. Minimal medium (CGXII [166]) charged with different carbon sources (cf. Table C.1) was infused at a rate of $300\,\mathrm{nL/min}$ with a high-precision syringe pump (neMESYS, Cetoni GmbH, Korbussen, Germany). The concentrations were adjusted to achieve equal molar concentration of carbon in each medium. PCA ($0.2\,\mathrm{mM}$) was used as chelating agent. A constant cultivation temperature of $30\,^{\circ}\mathrm{C}$ was ensured by an incubation chamber (PeCon GmbH, Erbach, Germany).

The microfluidic chip was mounted on a fully motorized inverted epifluorescence microscope (TI-Eclipse, Nikon GmbH, Düsseldorf, Germany) for time lapse imaging which was equipped with, the Nikon Perfect Focus System for thermal drift compensation, a Nikon Plan Apo 100 Ph3 DM Oil objective, the Nikon fluorescence excitation light

Table C.1: Carbon sources of MSCC experiments

Carbon source	Concentration
Glucose	$27\,\mathrm{mM}$ D-glucose
Gluconate	$27\,\mathrm{mM}$ D-gluconate
Fructose	$27\,\mathrm{mM}$ D-fructose
Acetate	$80\,\mathrm{mM}$ acetate
Pyruvate	$53\,\mathrm{mM}$ pyruvate
Citrate	$27\,\mathrm{mM}$ citrate + $CaCl_2$ ($5\,\mathrm{mM}$)

source (Intensilight), digital camera (Clara DR-3041, Andor Technology Plc., Belfast, United Kingdom) and LED light source (pE-100 white, CoolLed Ltd., Andover, UK).

C.2 Quantification Methods

Table C.2: Growth rates computed using different quantification methods for different datasets.

DatasetID	μ_{number} (95%)	μ_{area} (95%)	μ_{div}
16/2	0.804 (0.798, 0.810)	0.745 (0.743, 0.748)	0.798
03/1	0.809 (0.798, 0.821)	0.770 (0.766, 0.773)	0.811
33/2	0.822 (0.811, 0.832)	0.745 (0.742, 0.747)	0.788
35/1	0.816 (0.809, 0.823)	0.762 (0.758, 0.765)	0.806
39/2	0.821 (0.812, 0.831)	0.750 (0.746, 0.753)	0.813

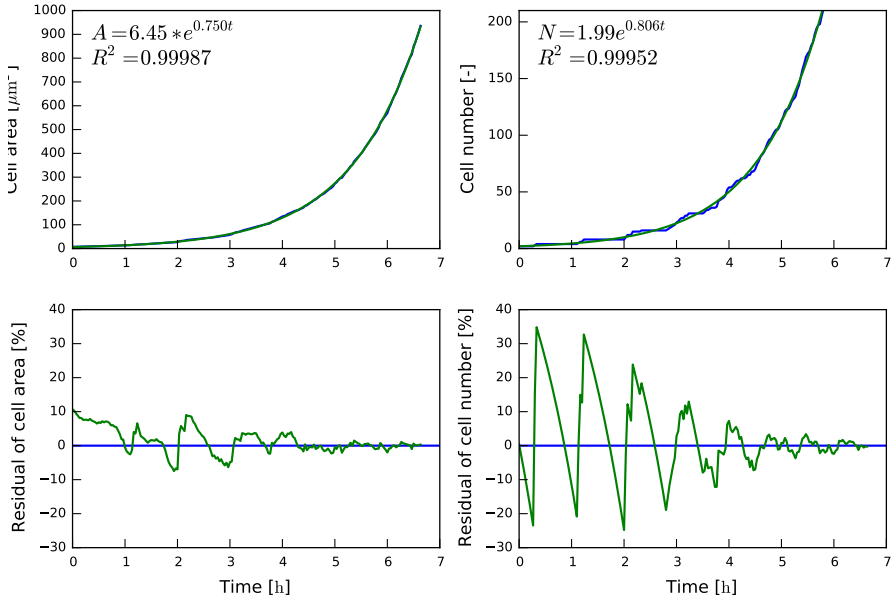


Figure C.1: Comparison between an exponential fit to the cumulative cell area of a population and the cell number of the same population (DatasetID 16/2). The blue line denotes measurement data, while the green line is the fit evaluated at the same positions as measurement data. The inlays show the fitted function with the optimal parameters and the R^2 value of the fit. The lower row shows residual plots for both quantification methods. The blue line denotes the exponential fit, the green line shows the deviation from experimental data in percent.

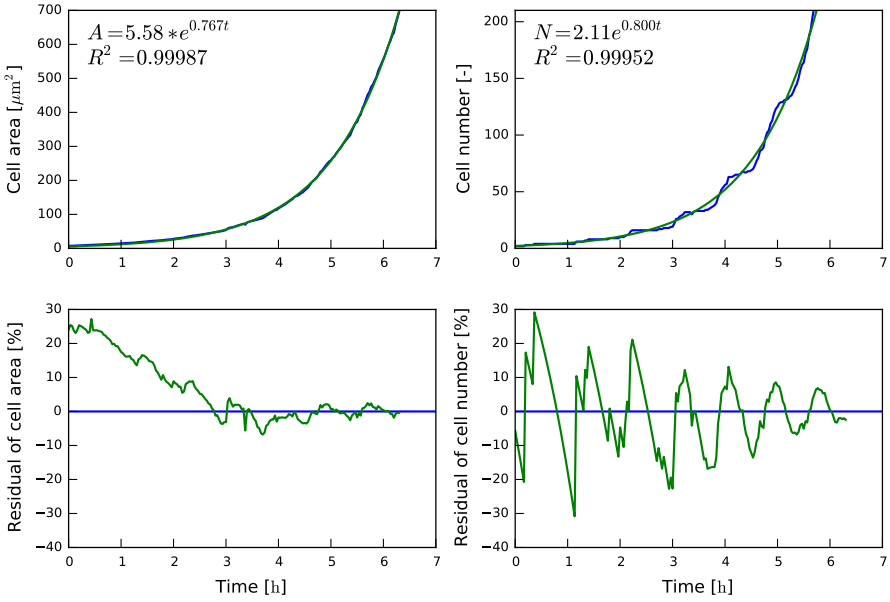


Figure C.2: Comparison between an exponential fit to the cumulative cell area of a population and the cell number of the same population (DatasetID 03/1). The blue line denotes measurement data, while the green line is the fit evaluated at the same positions as measurement data. The inlays show the fitted function with the optimal parameters and the R^2 value of the fit. The lower row shows residual plots for both quantification methods. The blue line denotes the exponential fit, the green line shows the deviation from experimental data in percent.

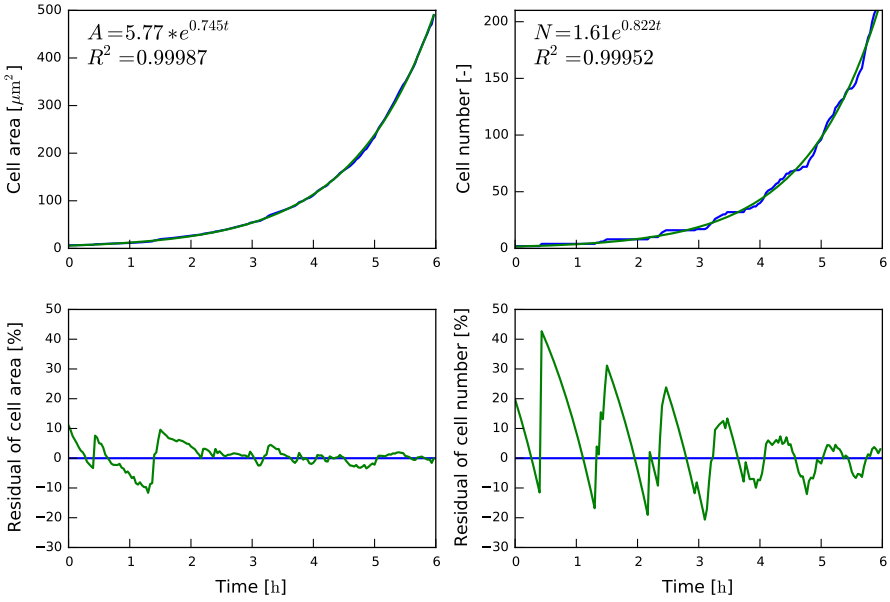


Figure C.3: Comparison between an exponential fit to the cumulative cell area of a population and the cell number of the same population (DatasetID 33/2). The blue line denotes measurement data, while the green line is the fit evaluated at the same positions as measurement data. The inlays show the fitted function with the optimal parameters and the R^2 value of the fit. The lower row shows residual plots for both quantification methods. The blue line denotes the exponential fit, the green line shows the deviation from experimental data in percent.

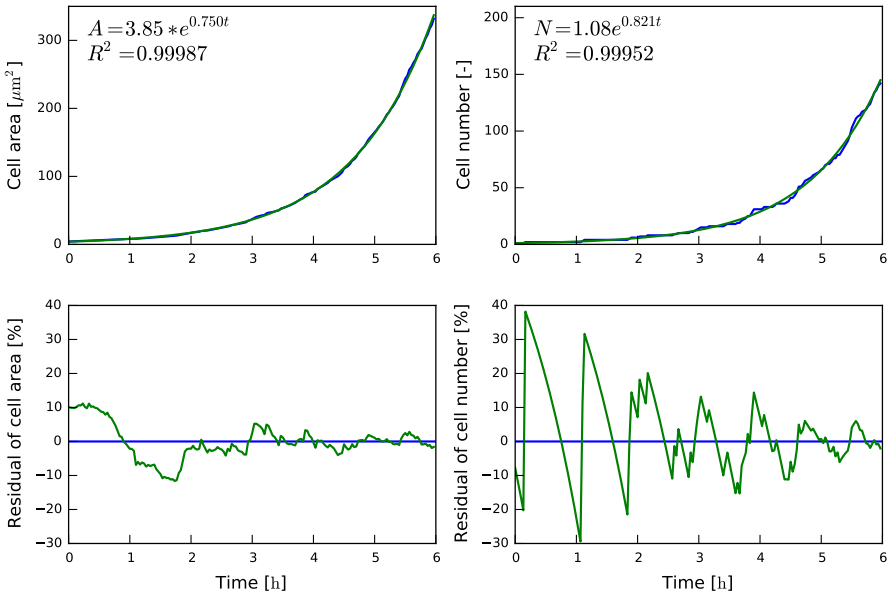


Figure C.4: Comparison between an exponential fit to the cumulative cell area of a population and the cell number of the same population (DatasetID 39/2). The blue line denotes measurement data, while the green line is the fit evaluated at the same positions as measurement data. The inlays show the fitted function with the optimal parameters and the R^2 value of the fit. The lower row shows residual plots for both quantification methods. The blue line denotes the exponential fit, the green line shows the deviation from experimental data in percent.

D Growth Parameter Correlations

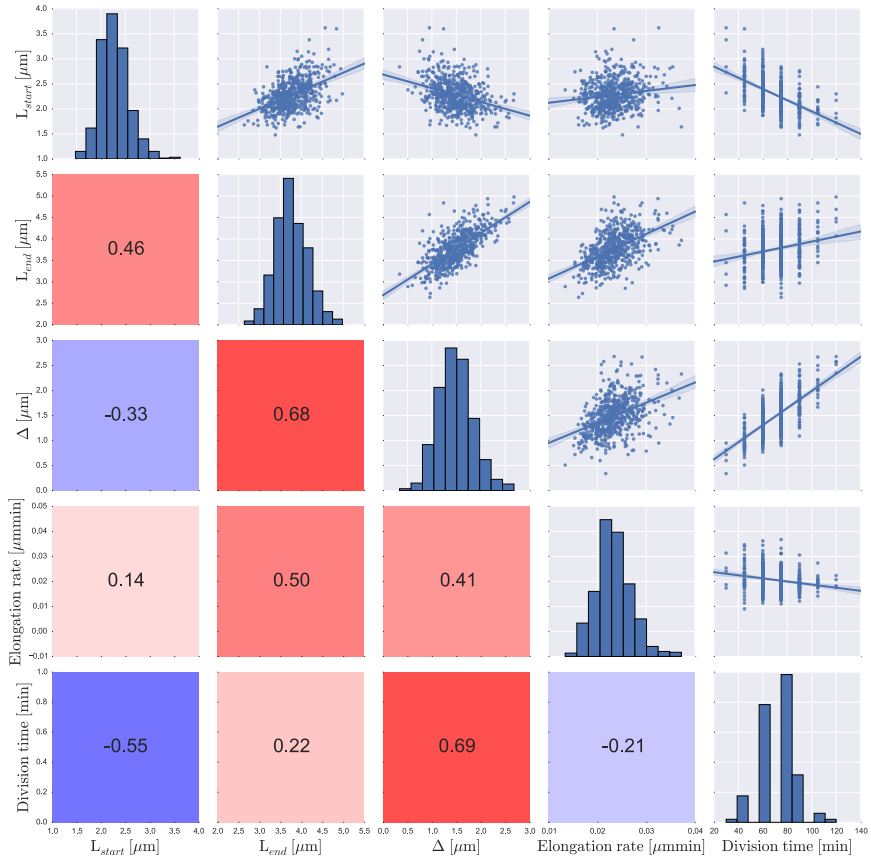


Figure D.1: Connection between single-cell growth and cell morphology in a correlation plot for phenotypic features based on cultivations with **acetate**. The values in the lower half of the matrix denote the Pearson correlation coefficient. Histograms for each characteristic are shown on the diagonal. The upper block of the matrix shows scatter plots of pairs of characteristics with a regression line and the 95% confidence band of a linear model.

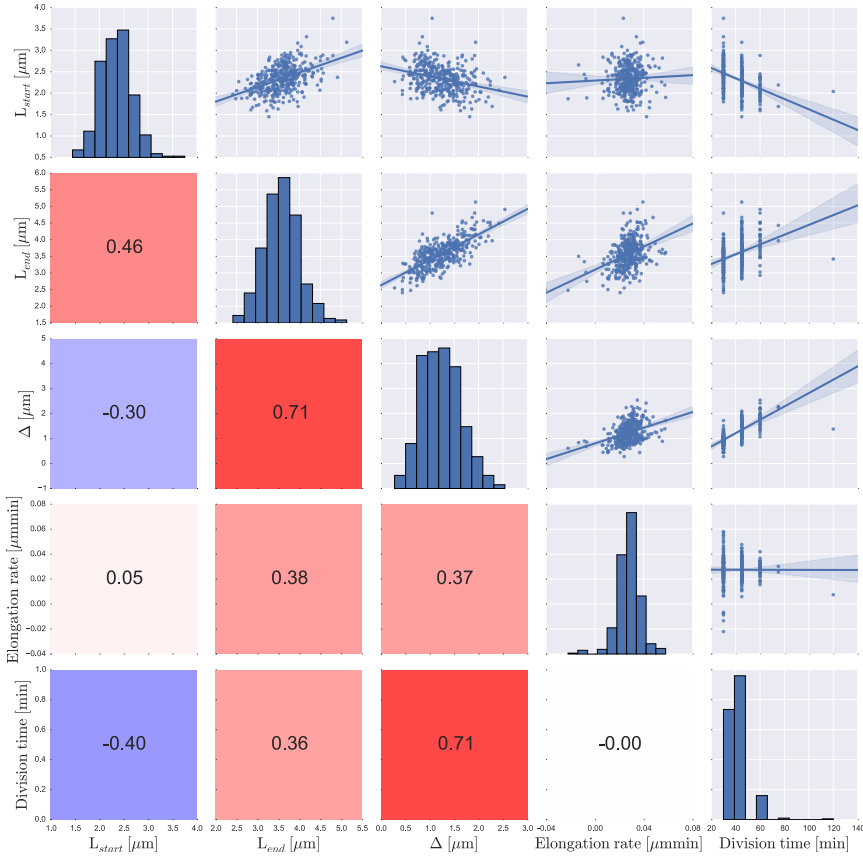


Figure D.2: Connection between single-cell growth and cell morphology in a correlation plot for phenotypic features based on cultivations with **citrate**. The values in the lower half of the matrix denote the Pearson correlation coefficient. Histograms for each characteristic are shown on the diagonal. The upper block of the matrix shows scatter plots of pairs of characteristics with a regression line and the 95% confidence band of a linear model.

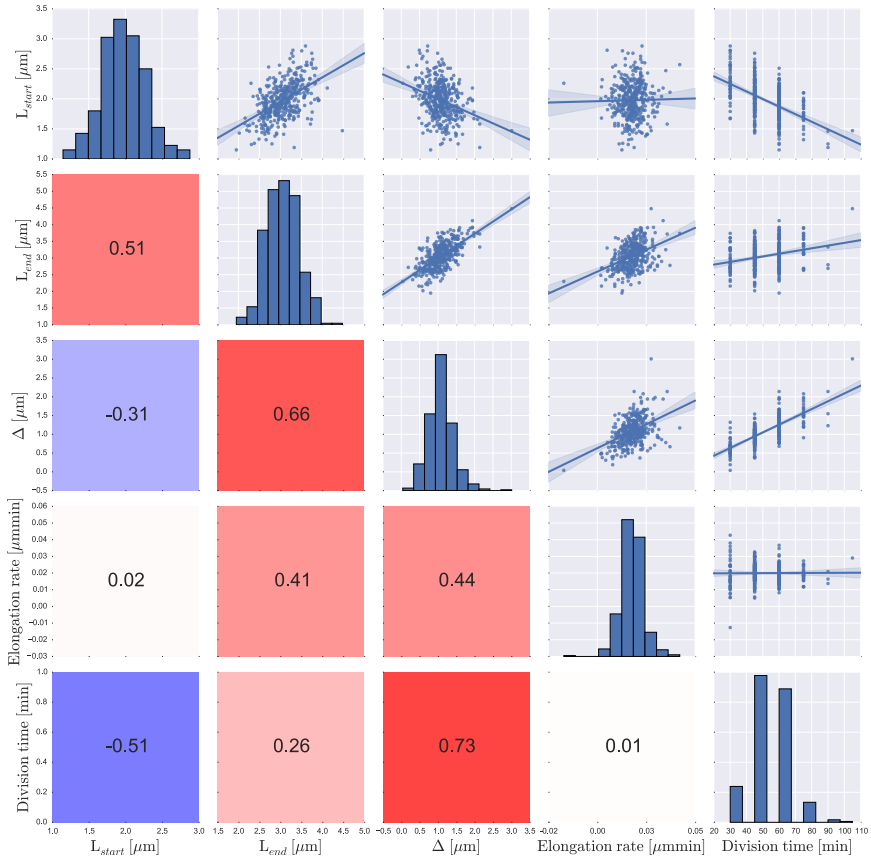


Figure D.3: Connection between single-cell growth and cell morphology in a correlation plot for phenotypic features based on cultivations with **fructose**. The values in the lower half of the matrix denote the Pearson correlation coefficient. Histograms for each characteristic are shown on the diagonal. The upper block of the matrix shows scatter plots of pairs of characteristics with a regression line and the 95% confidence band of a linear model.

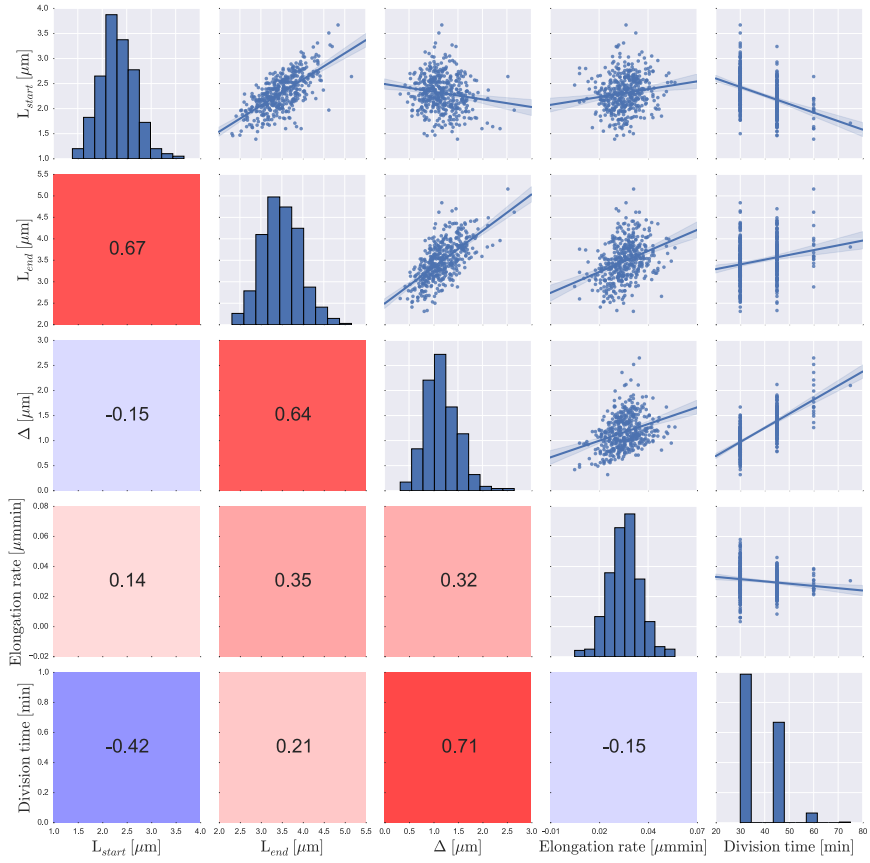


Figure D.4: Connection between single-cell growth and cell morphology in a correlation plot for phenotypic features based on cultivations with **gluconate**. The values in the lower half of the matrix denote the Pearson correlation coefficient. Histograms for each characteristic are shown on the diagonal. The upper block of the matrix shows scatter plots of pairs of characteristics with a regression line and the 95% confidence band of a linear model.

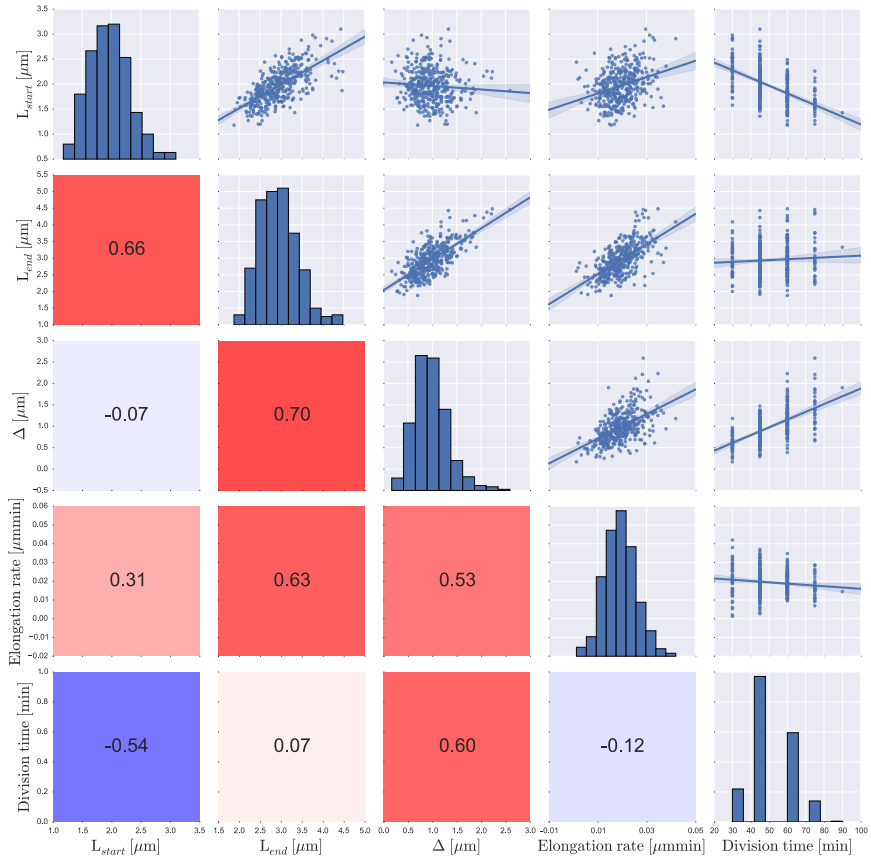


Figure D.5: Connection between single-cell growth and cell morphology in a correlation plot for phenotypic features based on cultivations with **glucose**. The values in the lower half of the matrix denote the Pearson correlation coefficient. Histograms for each characteristic are shown on the diagonal. The upper block of the matrix shows scatter plots of pairs of characteristics with a regression line and the 95% confidence band of a linear model.

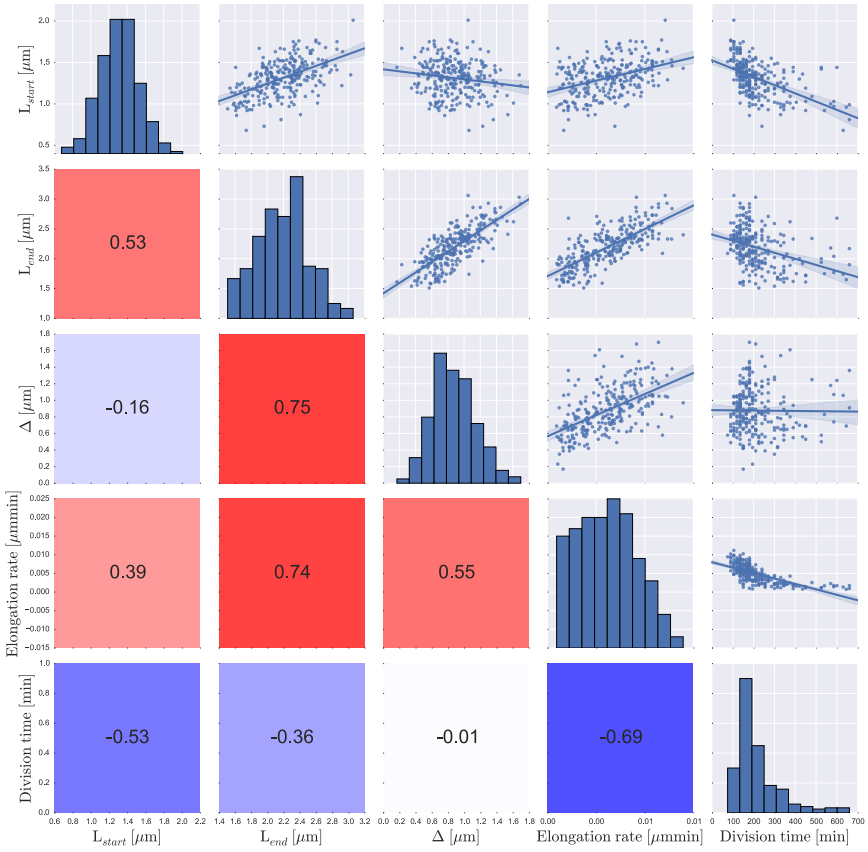


Figure D.6: Connection between single-cell growth and cell morphology in a correlation plot for phenotypic features based on cultivations with **PCA only**. The values in the lower half of the matrix denote the Pearson correlation coefficient. Histograms for each characteristic are shown on the diagonal. The upper block of the matrix shows scatter plots of pairs of characteristics with a regression line and the 95% confidence band of a linear model.

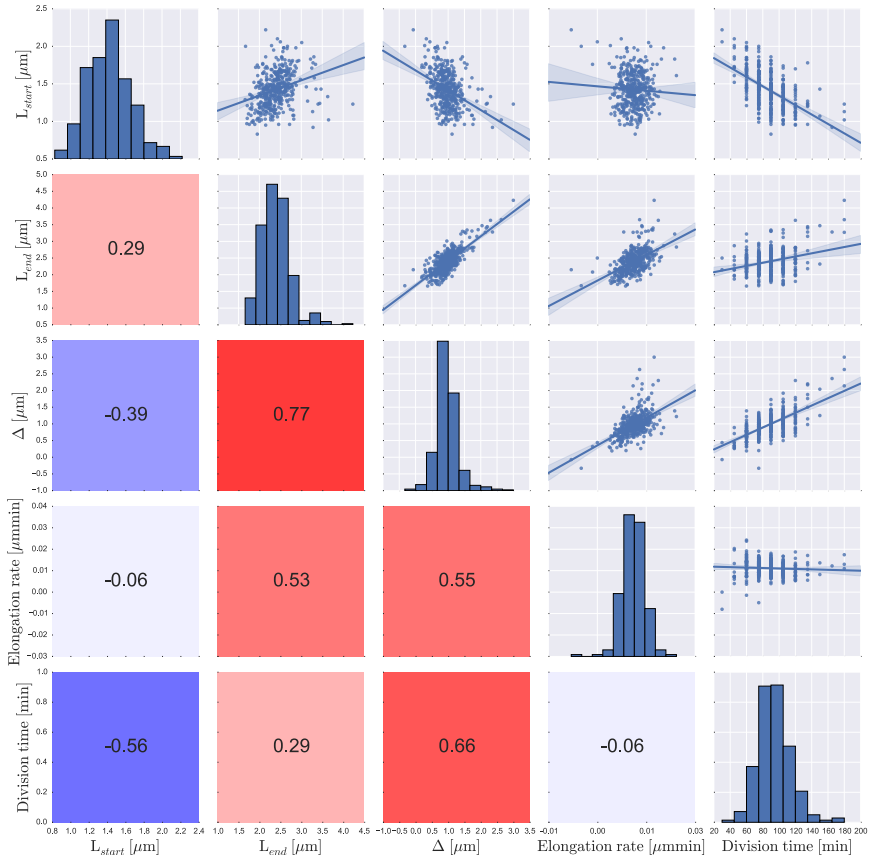


Figure D.7: Connection between single-cell growth and cell morphology in a correlation plot for phenotypic features based on cultivations with **pyruvate**. The values in the lower half of the matrix denote the Pearson correlation coefficient. Histograms for each characteristic are shown on the diagonal. The upper block of the matrix shows scatter plots of pairs of characteristics with a regression line and the 95% confidence band of a linear model.

E SOS/Phage Modeling

E.1 Experimental Procedures

E.1.1 Bacterial Strains and Growth Conditions

E. coli cells of the strain DH5α were cultivated in LB (Lysogeni Broth) medium or on agar plates at $37\,^{\circ}\mathrm{C}$. For growth studies and fluorescence based assays with *C. glutamicum* (e.g., preparation of cells for flow cytometry) cells were first cultivated in BHI (brain heart infusion, DifcoTM BHI, BD, Heidelberg, Germany) media at $30\,^{\circ}\mathrm{C}$ and were then used to inoculate a main culture in CGXII [166] with $2\,\%$ glucose. When necessary, $50\,\mu\mathrm{g m L}^{-1}$ (*E. coli*) or $25\,\mu\mathrm{g m L}^{-1}$ (*C. glutamicum*) kanamycin was added.

E.1.2 Cloning Techniques

Standard methods for example PCR or DNA restriction were performed according to established protocols [233]. Gibson assembly was used for plasmid construction [234]. DNA sequencing and oligonucleotides synthesis were conducted by Eurofins MWG Operon (Ebersberg, Germany). Plasmids and oligonucleotides used in this study are listed in Tables 1 and 2, respectively. The chromosomal integration of the SOS reporter (P_{recA}-*venus*) was performed using the two-step homologues recombination method [235]. Correct integration into the intergenic region of cg1121 and cg1122 was verified by colony PCR with the oligonucleotides Cg1121-Int-fw and Cg1122-Int-rv. Construction of the recA mutant of was performed analogously to the genomic integration by homologues recombination [235]. Correct deletion of the ATPase domain of recA was verified by colony PCR using the oligonucleotides D_recA_5 and D_recA_6.

E.1.3 Flow Cytometry

Flow cytometry analysis and cell sorting experiments were performed with a FACSAria II flow cytometer (BD, Heidelberg, Germany). A blue solid state laser with an excitation wavelength of $488\,\mathrm{nm}$ (to excite Venus) and a red gas laser for excitation at a wavelength of $633\,\mathrm{nm}$ (to excite E2-Crimson) were used. Cytometer set-up and

performance tracking was conducted with tracking beads labeled with a mixture of fluorochromes (BD, Heidelberg, Germany). Forward-scatter characteristics (FSC) and side-scatter characteristics (SSC) were detected as small and large angle scatters of the 488-nm laser. Fluorescence emitted by Venus was detected using a 502-nm long-pass and a 530/30-nm band pass filter set. E2-Crimson fluorescence was detected using a 660/20-nm band pass filter set. Analyses of cells were performed at a threshold rate of 2,000 to 5,000 events/s. Data were analyzed using FlowJo V10 (Tree Star, Inc., Ashland, OR).

E.1.4 Microfluidic Device Cultivation

An in-house developed microfluidic platform was used for *C. glutamicum* single-cell analysis [2, 59]. The microfluidic device incorporates a few hundred cultivation chambers with dimensions to ensure monolayer growth of isogenic microcolonies, with up to a few hundred cells maximum each. Phase contrast and fluorescence time-lapse imaging was performed at $8\,\mathrm{min}$ intervals.

Medium was supplied continuously to ensure stable and constant environmental conditions. Minimal medium (CGXII + $4\,\%$ glucose) with addition of $25\,\mu\mathrm{g/mL}$ kanamycin was infused at a rate of $300\,\mathrm{nL/min}$ with a high-precision syringe pump (neMESYS, Cetoni GmbH, Korbussen, Germany). For initiation of the starvation phase the medium was switched to minimal medium lacking glucose and protocatechuate (- carbon source) after an initial growth phase of $8\,\mathrm{h}$. Cells were exposed to carbon limitation for $\approx 24\,\mathrm{h}$. Then growth on standard CGXII minimal medium (+ carbon source) was resumed. A constant cultivation temperature of $30\,^{\circ}\mathrm{C}$ was ensured by an incubation chamber (PeCon GmbH, Erbach, Germany).

The microfluidic chip was mounted on a fully motorized inverted epifluorescence microscope (TI-Eclipse, Nikon GmbH, Düsseldorf, Germany) for time lapse imaging which was equipped with, the Nikon Perfect Focus System for thermal drift compensation, a Nikon Plan Apo 100 Ph3 DM Oil objective, the Nikon fluorescence excitation light source (Intensilight), digital cameras (Clara DR-3041 and Neo sCMOS, Andor Technology Plc., Belfast, United Kingdom) and LED light source (pE-100 white, CoolLed Ltd., Andover, UK). Optical filter blocks were installed for Venus fluorescence (EX 520/30 nm, DM 510 nm, EM 540/20 nm) and E2-Crimson fluorescence (EX 600/37 nm, DM 630 nm, EM 675/67 nm; AHF Analysentechnik AG, Tübingen, Germany).

E.1.5 Image Analysis and Data Visualization

Time-lapse movies of monolayer growth chambers were analyzed using a custom, specialized workflow implemented as an ImageJ/Fiji plugin [125]. Cell identification was performed using a segmentation procedure tailored to detect individual rod-shaped cells in crowded populations using an advanced Watershed-type approach. Detected cells are subsequently tracked throughout image sequences using the single particle

tracking approach [115] as implemented in TrackMate [122]. The image analysis allows the extraction of measurable quantities of individual cells, i.e. mean fluorescence intensities, cell area, as well as derived quantities (i.e., growth rate). Mother machine image analysis was performed using custom software tuned for fast extraction of cell information from 1D growth channels [236]. All data sets were subsequently processed using the analysis and visualization software *Vizardous* [129].

E.2 Trigger of Phage Induction

History and future of SOS+ cells. Here, we performed an analysis of the recorded lineage data in order to compare the growth of SOS-induced with uninduced cells even before the induction event. Do cells which will suffer from SOS induction in the future show an altered growth already before the induction of the stress response? For example, are fast growing cells even more prone to spontaneous SOS? Or in contrast, do we even see a reduction in growth rate since the cell is already suffering from e. g. oxidative stress?

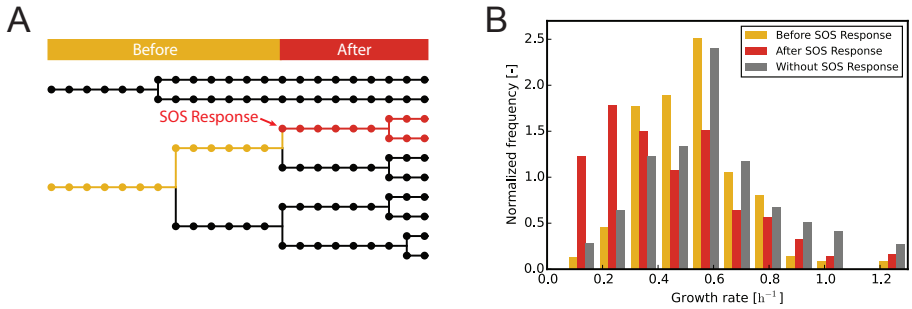


Figure E.1: A. Schematic lineage tree illustrating the dissection of single traces in a "before" and "after" state upon SOS induction. **B.** Distribution of the growth rate of cells "before" and "after" SOS induction. These values are compared to cells which did not exhibit SOS induction (gray bars) throughout the experiment. Growth rates μ were computed as $\mu = \frac{\ln(2)}{t_{gen}}$. For complete generations, t_{gen} is defined as the time between two cell divisions. For incomplete generations, the end of the experiment is defined as cell division. Consequently, a growth rate $>0\,\mathrm{h}^{-1}$ is computed even for non-growing cells.

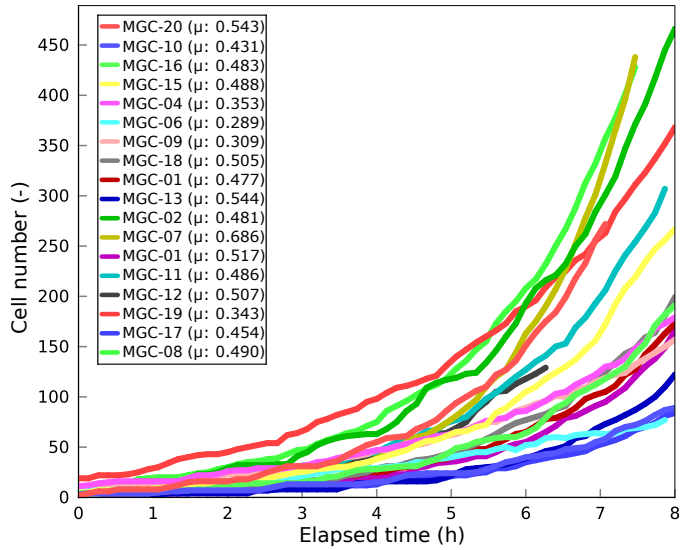


Figure E.2: Cells were grown in microfluidic chip device in CGXII minimal medium with $2\,\%$ glucose (for a detailed description, see material and methods). An average growth rate of $0.47\,\mathrm{h}^{-1}$ was observed.

E.3 Setting of Thresholds for Fluorescence Data

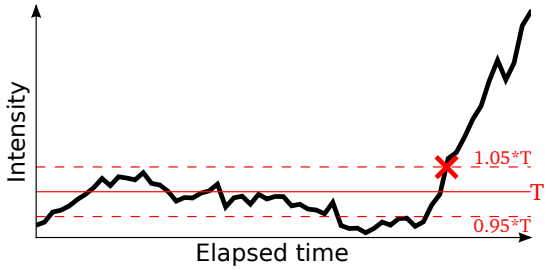


Figure E.3: Biological as well as technical noise may result in small fluctuations of fluorescence reporter outputs. Applying a single threshold would result in many false-positive state transitions when the signal fluctuates around the threshold. To prevent such effects, an *UNDEFINED* state around the selected threshold (T) was established (between $95\,\%$ and $105\,\%$ of T) in which the temporal context of a cell is taken into account to determine the cells' correct state: Cells that enter but leave the *UNDEFINED* state to an OFF state are not counted as entering the ON state. A cell is assigned the ON state at the time point where the fluorescence reporter intensity exceeds the upper threshold of the *UNDEFINED* state. The same applies for cells entering the *UNDEFINED* state from the ON state and for the lower

threshold of the *UNDEFINED* state.

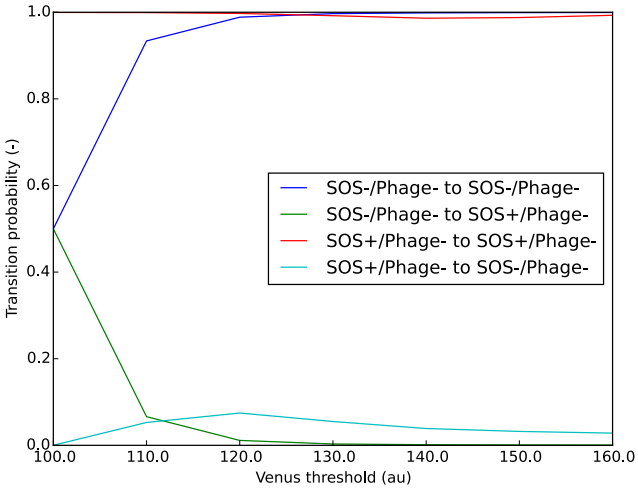


Figure E.4: Appropriate thresholds for Venus (T_{SOS}) and E2-Crimson (T_{Phage}) intensities were set, for both reporters separately, to approximately half the maximum value of observed intensities. T_{SOS} is more than 3 standard deviations above the mean of the SOS uninduced population (Mean: $117.31\,\mathrm{h}^{-1}$, StdDev: $3.85\,\mathrm{h}^{-1}$). T_{Phage} has been set empirically in a way that cells that exceed the threshold do not generate offspring. Furthermore, we have conducted a study about the influence of T_{SOS} on transition probabilities.

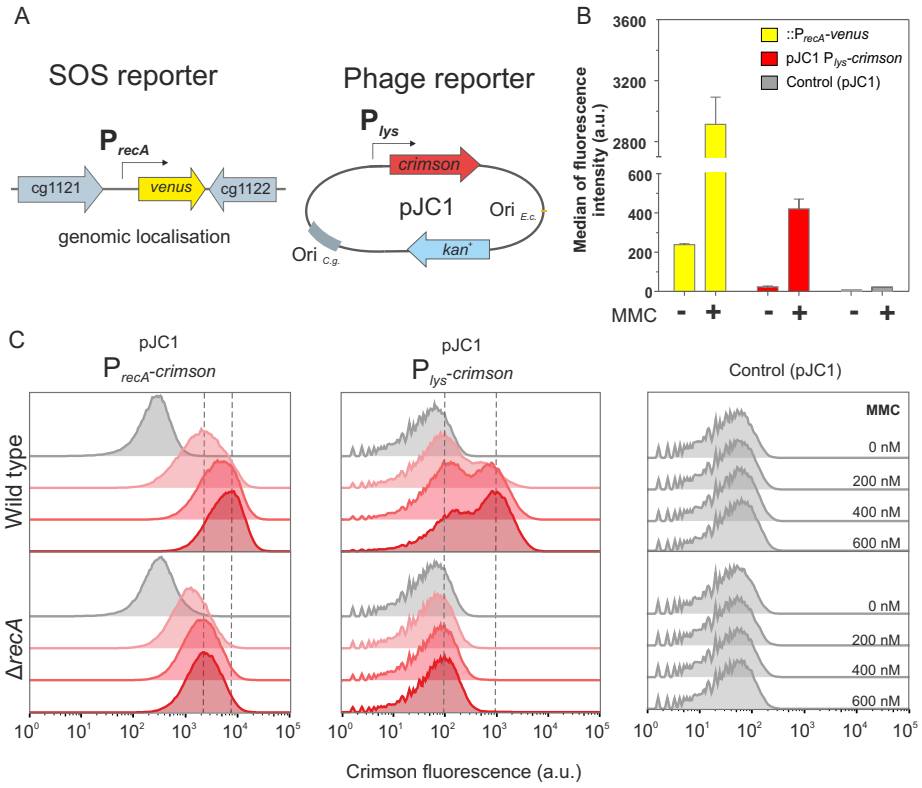


Figure E.5: Construction and verification of the reporter strain. **A.** The reporter strain used in the following studies carries a genomically integrated SOS (P_{recA}-*venus*) and a plasmid-based phage reporter (P_{lys}-*crimson*) construct. **B.** The dual reporter strain shows a strongly increased activity of both reporters after six hours cultivation in glucose minimal medium upon addition of $600\,\mathrm{nmol}$ Mitomycin C (MMC) in comparison to the cells without MMC. **C.** Comparison of the output of the SOS and phage reporter in wild type and ΔrecA cells containing the plasmid pJC1-P_{recA}-*crimson*, pJC1-P_{lys}-*crimson*, or the empty vector control after six hours cultivation in glucose minimal medium upon addition of different concentrations of MMC.

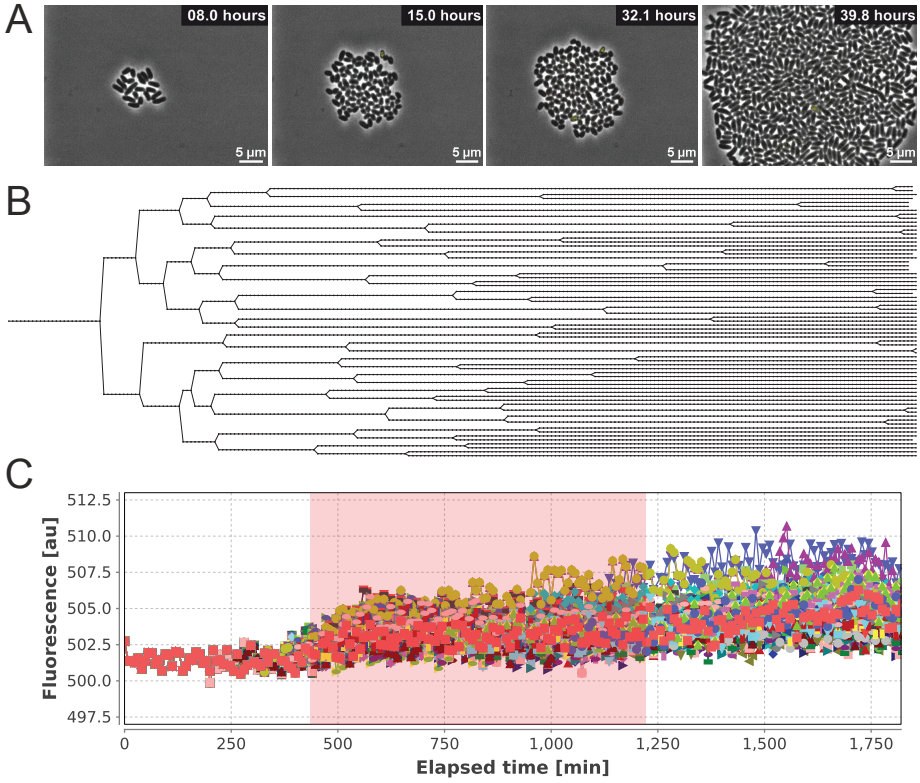


Figure E.6: Nutrient limitation case study. A. Images extracted from the experiment showing a growing population with experimentally restricted growth between $8\,\mathrm{h}$ and $24\,\mathrm{h}$. **B.** Excerpt of the lineage tree from the nutrient limitation experiment. **C.** Time-resolved data for the fluorescence reporter processed with *Vizardous*. The red rectangle denotes the nutrient limitation phase. Regular growth phases with fast division patterns seen in the lineage tree are characterized by low fluorescence intensities in the first $8\,\mathrm{h}$ in comparison to the starvation phase. Apparently, fluorescent signals as well as their heterogeneity are increasing after this time. Notably, these signals not decrease after nutrients are again available.

E.4 Population Modeling

Description	Reaction	Parameter
SOS^-Phage^- to SOS^+Phage^-	$S_0P_0 \longrightarrow S_1P_0$	$k_1 = 0.0001$
SOS^+Phage^- to SOS^-Phage^-	$S_1P_0 \longrightarrow S_0P_0$	$k_2 = 0.001$
SOS^+Phage^- to SOS^+Phage^+	$S_1P_0 \longrightarrow S_1P_1$	$k_3 = 0.002$
SOS^+Phage^+ to SOS^-Phage^+ (inactive)	$S_1P_1 \longrightarrow S_0P_1$	$k_4 = 0.001$
SOS^-Phage^- to SOS^-Phage^+	$S_0P_0 \longrightarrow S_0P_1$	$k_5 = 0.00001$
SOS^-Phage^- doubling	$S_0P_0 \longrightarrow 2 \cdot S_0P_0$	$k_6 = 0.00045$
SOS^+Phage^- doubling	$S_1P_0 \longrightarrow 2 \cdot S_1P_0$	$k_7 = 0.0003$
SOS^+Phage^+ removed	$S_1P_1 \longrightarrow \emptyset$	$k_8 = 0.005$
SOS^-Phage^+ removed	$S_0P_1 \longrightarrow \emptyset$	$k_9 = 0.005$

Table E.1: Reactions that describe the regulatory scheme of Figure 12.2 in a stochastic simulation framework. Reactions follow the law of mass action and depend on the amount of reactants and the associated parameters.

E.5 Molecular Modeling of SOS and Phage Dynamics

Description	Reaction	Propensity
recA mRNA transcription	$f_r \longrightarrow f_r + m_r$	$gmr \cdot (1 - br)$
lexA mRNA transcription	$f_l \longrightarrow f_l + m_l$	$gml \cdot (1 - bl)$
RecA protein translation	$m_r \longrightarrow m_r + r$	gr
LexA protein translation	$m_l \longrightarrow m_l + l$	gl
recA mRNA degradation	$m_r \longrightarrow \emptyset$	dmr
lexA mRNA degradation	$m_l \longrightarrow \emptyset$	dml
RecA protein degradation	$r \longrightarrow \emptyset$	dr
LexA protein degradation	$l \longrightarrow \emptyset$	dl
LexA binding to *recA* promoter	$l + f_r \longrightarrow b_r$	$cr \cdot (1 - br) \cdot l$
LexA binding to *lexA* promoter	$l + f_l \longrightarrow b_l$	$cl \cdot (1 - bl) \cdot l$
LexA binding to *x* promoter	$l + f_x \longrightarrow b_x$	$cx \cdot (1 - bx) \cdot l$
LexA dissociation from *recA*	$b_r \longrightarrow f_r + l$	ur
LexA dissociation from *lexA*	$b_l \longrightarrow f_l + l$	ul
LexA dissociation from X	$b_x \longrightarrow f_x + l$	ux
RecA marks LexA for cleavage	$r + l \longrightarrow r$	cp
eyfp mRNA transcription	$f_g \longrightarrow f_g + m_g$	$gmy \cdot (n - by)$
eYFP translation	$m_g \longrightarrow m_g + g$	gy
eyfp mRNA degradation	$m_g \longrightarrow \emptyset$	dmy
LexA binding to *eyfp* promoter	$l + f_g \longrightarrow b_g$	$cy \cdot (n - by) \cdot l$
LexA dissociation from *eyfp* promoter	$b_g \longrightarrow f_g + l$	uy
eYFP degradation	$g \longrightarrow \emptyset$	dy
X binding to *int* promoter	$x + f_i \longrightarrow b_i$	$ci \cdot (1 - bi) \cdot x$
X dissociation from *int* promoter	$b_i \longrightarrow f_i + x$	ui
int mRNA transcription	$f_i \longrightarrow f_i + m_i$	$gmi \cdot (1 - fi)$
Int protein translation	$m_i \longrightarrow m_i + i$	gi
int mRNA degradation	$m_i \longrightarrow \emptyset$	dmi
Int protein degradation	$i \longrightarrow \emptyset$	di
crimson mRNA transcription	$b_c \longrightarrow b_c + m_c$	$gmc \cdot (m - fc)$
Crimson translation	$m_c \longrightarrow m_c + c$	gc
crimson mRNA degradation	$m_c \longrightarrow \emptyset$	dmc
X binding to *crimson* promoter	$x + f_c \longrightarrow b_c$	$cc \cdot (m - bc) \cdot x$
X dissociation from *crimson* promoter	$b_c \longrightarrow f_c + x$	uc
Crimson degradation	$c \longrightarrow \emptyset$	dc
x mRNA transcription	$f_x \longrightarrow f_x + m_x$	$gmx \cdot (1 - bx)$
X translation	$m_x \longrightarrow m_x + x$	gx
x mRNA degradation	$m_x \longrightarrow \emptyset$	dmx
X degradation	$x \longrightarrow \emptyset$	dx

Table E.2: Reactions that describe the regulatory scheme of Figure 12.7. Each reaction is furthermore coupled to a parameter that is defined in Table E.3. Initial values for reactants (species) are provided in Table E.4.

Parameter	Value
gmr	0.05
gml	0.03
gr	0.04
gl	0.1
dmr	0.02
dml	0.003
dr	0.02
dl	0.002
cr	0.02
cl	0.01
ur	0.04
ul	0.01
cp	0.0001
gmy	0.15
gy	0.08
dmy	0.3
cy	0.05
uy	0.04
dy	0.00025
ci	0.05
ui	0.1
gmi	0.1
gi	0.02
dmi	0.05
di	0.0002
cc	0.2
uc	0.2
gmc	0.05
gc	0.2
dmc	0.2
dc	0.01
cx	0.05
ux	0.05
gmx	0.05
gx	0.2
dmx	0.02
dx	0.03

Table E.3: Parameters used for the simulations that are depicted in Figure 12.9.

Description	Variable	Initial value
Free *recA* promoter	f_r	0
recA mRNA	m_r	0
Free *lexA* promoter	f_l	0
lexA mRNA	m_l	0
RecA	r	1
LexA	l	25
Bound *recA* promoter	b_r	1
Bound *lexA* promoter	b_l	1
Free *yfp* promoter	f_y	0
yfp mRNA	m_y	0
Bound *yfp* promoter	b_y	1
YFP	y	0
Free *integrase* promoter	f_i	1
Bound *integrase* promoter	b_i	0
integrase mRNA	m_i	0
Integrase	i	0
Free *crimson* promoter	f_c	10
Bound *crimson* promoter	b_c	0
mRNA *crimson*	m_c	0
Crimson	c	0
Free *x* promoter	f_x	0
Bound *x* promoter	b_x	1
x mRNA	m_x	0
X	x	0

Table E.4: Initial values for reactants (species) of the reactions described in Table E.2.

Bibliography

[1] S. V. Avery: Microbial Cell Individuality and the Underlying Sources of Heterogeneity. *Nature Reviews Microbiology*, **4**:8 (2006), pp. 577–87. doi: 10.1038/nrmicro1460. (see pp. v–vi).

[2] A. Grünberger, J. van Ooyen, N. Paczia, P. Rohe, G. Schiendzielorz, L. Eggeling, W. Wiechert, D. Kohlheyer, and S. Noack: Beyond growth rate 0.6: *Corynebacterium glutamicum* cultivated in highly diluted environments. *Biotechnology and Bioengineering*, **110**:1 (2013), pp. 220–228. doi: 10.1002/bit.24616. (see pp. v–vi, 3–4, 15–16, 83, 119, 167, 182).

[3] C. D. Murphy: The microbial cell factory. *Organic & Biomolecular Chemistry*, **10**:10 (2012), p. 1949. doi: 10.1039/c2ob06903b (see p. 3).

[4] D. C. Williams, R. M. Van Frank, W. L. Muth, and J. P. Burnett: Cytoplasmic inclusion bodies in *Escherichia coli* producing biosynthetic human insulin proteins. *Science (New York, N.Y.)*, **215**:4533 (1982), pp. 687–689. doi: 10.1126/science.7036343 (see p. 3).

[5] P. Jeandet, Y. Vasserot, T. Chastang, and E. Courot: Engineering microbial cells for the biosynthesis of natural compounds of pharmaceutical significance. *BioMed Research International*, **2013**: (2013), pp. 1–35. doi: 10.1155/2013/780145 (see p. 3).

[6] J. Becker and C. Wittmann: Bio-based production of chemicals, materials and fuels - *Corynebacterium glutamicum* as versatile cell factory. *Current Opinion in Biotechnology*, **23**:4 (2012), pp. 631–640. doi: 10.1016/j.copbio.2011.11.012. (see p. 3).

[7] C. Dellomonaco, F. Fava, and R. Gonzalez: The path to next generation biofuels: successes and challenges in the era of synthetic biology. *Microbial Cell Factories*, **9**: (2010), p. 3. doi: 10.1186/1475-2859-9-3 (see p. 3).

[8] B. Blombach, T. Riester, S. Wieschalka, C. Ziert, J. W. Youn, V. F. Wendisch, and B. J. Eikmanns: *Corynebacterium glutamicum* tailored for efficient isobutanol production. *Applied and Environmental Microbiology*, **77**:10 (2011), pp. 3300–3310. doi: 10.1128/AEM.02972-10 (see pp. 3, 10).

[9] G. Festel, C. Detzel, and R. Maas: Industrial biotechnology - Markets and industry structure. 2012. doi: 10.5912/jcb.478. (see p. 3).

[10] Innovationsinitiative industrielle Biotechnologie. url: http://www.bmbf.de/de/16336.php (visited on 05/12/2015) (see p. 3).

[11] Industrial Biotechnology (IB) Catalyst. url: http://www.bbsrc.ac.uk/business/collaborative-research/innovate-uk-competitions/ib-catalyst/ (visited on 05/12/2015) (see p. 3).

[12] National Research Strategy BioEconomy 2030. url: http://www.bmbf.de/de/biooekonomie.php (visited on 05/27/2015) (see p. 3).

[13] K. Buchholz and J. Collins: The roots–a short history of industrial microbiology and biotechnology. *Applied Microbiology and Biotechnology*, **97**:9 (2013), pp. 3747–3762. doi: 10.1007/s00253-013-4768-2 (see p. 3).

[14] D. J. Kiviet, P. Nghe, N. Walker, S. Boulineau, V. Sunderlikova, and S. J. Tans: Stochasticity of metabolism and growth at the single-cell level. *Nature*, (2014). doi: 10.1038/nature13582. (see pp. 3, 111).

[15] A. Grünberger, W. Wiechert, and D. Kohlheyer: Single-cell microfluidics: Opportunity for bioprocess development. 2014. doi: 10.1016/j.copbio.2014.02.008. (see pp. 3–4, 14, 83, 99, 101).

[16] S. Unthan, A. Grünberger, J. van Ooyen, J. Gätgens, J. Heinrich, N. Paczia, W. Wiechert, D. Kohlheyer, and S. Noack: Beyond growth rate 0.6: What drives *Corynebacterium glutamicum* to higher growth rates in defined medium. *Biotechnology and Bioengineering*, **111**:2 (2014), pp. 359–371. doi: 10.1002/bit.25103 (see pp. 3, 108).

[17] M. E. Lidstrom and M. C. Konopka: The Role of Physiological Heterogeneity in Microbial Population Behavior. *Nature Chemical Biology*, **6**:10 (2010), pp. 705–12. doi: 10.1038/nchembio.436. (see p. 3).

[18] N. Rosenfeld, J. W. Young, U. Alon, P. S. Swain, and M. B. Elowitz: Gene Regulation at the Single-Cell Level. *Science*, **307**:5717 (2005), pp. 1962–5. doi: 10.1126/science.1106914. (see p. 3).

[19] A. Mader, B. von Bronk, B. Ewald, S. Kesel, K. Schnetz, E. Frey, and M. Opitz: Amount of Colicin Release in *Escherichia coli* Is Regulated by Lysis Gene Expression of the Colicin E2 Operon. *PLoS ONE*, **10**:3 (2015), e0119124. doi: 10.1371/journal.pone.0119124. (see p. 3).

[20] T. Lu, T. Shen, M. R. Bennett, P. G. Wolynes, and J. Hasty: Phenotypic variability of growing cellular populations. *Proceedings of the National Academy of Sciences of the United States of America*, **104**:48 (2007), pp. 18982–18987. doi: 10.1073/pnas.0706115104 (see p. 3).

[21] K. P. Koutsoumanis and A. Lianou: Stochasticity in colonial growth dynamics of individual bacterial cells. *Applied and Environmental Microbiology*, January (2013). doi: 10.1128/AEM.03629-12. (see p. 3).

[22] G. Fritz, J. a. Megerle, S. a. Westermayer, D. Brick, R. Heermann, K. Jung, J. O. Rädler, and U. Gerland: Single cell kinetics of phenotypic switching in the arabinose utilization system of *E. coli*. *PLoS ONE*, **9**:2 (2014). doi: 10.1371/journal.pone.0089532 (see p. 3).

[23] S. Boulineau, F. Tostevin, D. J. Kiviet, P. R. ten Wolde, P. Nghe, and S. J. Tans: Single-Cell Dynamics Reveals Sustained Growth during Diauxic Shifts. *PLoS ONE*, **8**:4 (2013). doi: 10.1371/journal.pone.0061686 (see p. 3).

[24] S. Lindström and H. Andersson-Svahn: Overview of single-cell analyses: microdevices and applications. *Lab on a Chip*, **10**:24 (2010), pp. 3363–3372. doi: 10.1039/c0lc00150c (see p. 4).

[25] B. Okumus, S. Yildiz, and E. Toprak: Fluidic and microfluidic tools for quantitative systems biology. *Current Opinion in Biotechnology*, **25**: (2014), pp. 30–38. doi: 10.1016/j.copbio.2013.08.016. (see pp. 4, 99).

[26] N. Mustafi, A. Grünberger, R. Mahr, S. Helfrich, K. Nöh, B. Blombach, D. Kohlheyer, and J. Frunzke: Application of a genetically encoded biosensor for live cell imaging of L-valine production in pyruvate dehydrogenase complex-deficient *Corynebacterium glutamicum* strains. *PLoS ONE*, **9**:1 (2014), e85731. doi: 10.1371/journal.pone.0085731. (see pp. 4, 12, 93, 150, 215).

[27] H. Hutter: Fluorescent reporter methods. *Methods in Molecular Biology (Clifton, N.J.)*, **351**: (2006), pp. 155–173. doi: 10.1385/1-59745-151-7:155 (see p. 4).

[28] M. B. Elowitz, A. J. Levine, E. D. Siggia, and P. S. Swain: Stochastic gene expression in a single cell. *Science*, **297**:5584 (2002), pp. 1183–1186 (see p. 4).

[29] A. Novick and M. Weiner: Enzyme induction as an all-or-none Phenomenon. *Proceedings of the National Academy of Sciences of the United States of America*, **43**:7 (1957), pp. 553–566 (see p. 4).

[30] L. Eggeling and M. Bott: Handbook of *Corynebacterium glutamicum*. Ed. by L. Eggeling and M. Bott. Boca Raton, FL: CRC Press, Taylor & Francis Group, 2005. (see pp. 10–12, 118).

[31] M. Inui, S. Murakami, S. Okino, H. Kawaguchi, A. a. Vertès, and H. Yukawa: Metabolic analysis of *Corynebacterium glutamicum* during lactate and succinate productions under oxygen deprivation conditions. *Journal of Molecular Microbiology and Biotechnology*, **7**:4 (2004), pp. 182–196. doi: 10.1159/000079827 (see p. 10).

[32] R. a. Weusthuis, I. Lamot, J. van der Oost, and J. P. M. Sanders: Microbial production of bulk chemicals: Development of anaerobic processes. *Trends in Biotechnology*, **29**:4 (2011), pp. 153–158. doi: 10.1016/j.tibtech.2010.12.007. (see p. 10).

[33] A. McGuire, B. Weiner, S. Park, I. Wapinski, S. Raman, G. Dolganov, M. Peterson, R. Riley, J. Zucker, T. Abeel, J. White, P. Sisk, C. Stolte, M. Koehrsen, R. T. Yamamoto, M. Iacobelli-Martinez, M. J. Kidd, A. M. Maer, G. K. Schoolnik, A. Regev, and J. Galagan: Comparative analysis of mycobacterium and related actinomycetes yields insight into the evolution of *Mycobacterium tuberculosis* pathogenesis. *BMC Genomics*, **13**:1 (2012), p. 120. doi: 10.1186/1471-2164-13-120. (see p. 10).

[34] H. Yukawa and M. Inui: *Corynebacterium glutamicum*: Biology and Biotechnology. Ed. by H. Yukawa and M. Inui. Springer Berlin / Heidelberg, 2013. doi: 10.1007/978-3-642-29857-8 (see pp. 10, 118).

[35] A. Burkovski, ed.: *Corynebacterium glutamicum*: From Systems Biology to Biotechnological Applications. Caister Academic Press, 2015, pp. 1–198 (see p. 10).

[36] C. Dusny, A. Grünberger, C. Probst, W. Wiechert, D. Kohlheyer, and A. Schmid: Technical bias of microcultivation environments on single-cell physiology. *Lab on a Chip*, **15**: (2015), pp. 1822–1834. doi: 10.1039/C4LC01270D. (see pp. 10, 108, 111).

[37] R. a. Daniel and J. Errington: Control of cell morphogenesis in bacteria: Two distinct ways to make a rod-shaped cell. *Cell*, **113**:6 (2003), pp. 767–776. doi: 10.1016/S0092-8674(03)00421-5 (see pp. 10–11).

[38] C. Donovan and M. Bramkamp: Cell division in Corynebacterineae. *Frontiers in Microbiology*, **5**:132 (2014), pp. 1–16. doi: 10.3389/fmicb.2014.00132 (see pp. 10, 127).

[39] M. Bramkamp and S. van Baarle: Division site selection in rod-shaped bacteria. *Current Opinion in Microbiology*, **12**:6 (2009), pp. 683–688. doi: 10.1016/j.mib.2009.10.002 (see p. 10).

[40] N. R. Thanky, D. B. Young, and B. D. Robertson: Unusual features of the cell cycle in mycobacteria: Polar-restricted growth and the snapping-model of cell division. *Tuberculosis*, **87**:3 (2007), pp. 231–236. doi: 10.1016/j.tube.2006.10.004 (see p. 11).

[41] E. C. Hett and E. J. Rubin: Bacterial growth and cell division: a mycobacterial perspective. *Microbiology and Molecular Biology Reviews*, **72**:1 (2008), 126–156, table of contents. doi: 10.1128/MMBR.00028-07 (see p. 11).

[42] M. Letek, M. Fiuza, E. Ordóñez, A. F. Villadangos, A. Ramos, L. M. Mateos, and J. a. Gil: Cell growth and cell division in the rod-shaped actinomycete *Corynebacterium glutamicum*. *Antonie van Leeuwenhoek*, **94**:1 (2008), pp. 99–109. doi: 10.1007/s10482-008-9224-4 (see p. 11).

[43] J. Kalinowski, B. Bathe, D. Bartels, N. Bischoff, M. Bott, A. Burkovski, N. Dusch, L. Eggeling, B. J. Eikmanns, L. Gaigalat, A. Goesmann, M. Hartmann, K. Huthmacher, R. Krämer, B. Linke, A. C. McHardy, F. Meyer, B. Möckel, W. Pfefferle, A. Pühler, D. a. Rey, C. Rückert, O. Rupp, H. Sahm, V. F. Wendisch, I. Wiegräbe, and A. Tauch: The complete *Corynebacterium glutamicum* ATCC 13032 genome sequence and its impact on the production of l-aspartate-derived amino acids and vitamins. *Journal of Biotechnology*, **104**:1-3 (2003), pp. 5–25. doi: 10.1016/S0168-1656(03)00154-8. (see pp. 11–12).

[44] M. Ikeda and S. Nakagawa: The *Corynebacterium glutamicum* genome: Features and impacts on biotechnological processes. *Applied Microbiology and Biotechnology*, **62**:2-3 (2003), pp. 99–109. doi: 10.1007/s00253-003-1328-1 (see p. 11).

[45] A. B. Oppenheim, O. Kobiler, J. Stavans, D. L. Court, and S. Adhya: Switches in Bacteriophage Lambda Development. *Annual Review of Genetics*, **39**: (2005), pp. 409–29. doi: 10.1146/annurev.genet.39.073003.113656. (see pp. 11–13, 129).

[46] A. M. Nanda, A. Heyer, C. Krämer, A. Grünberger, D. Kohlheyer, and J. Frunzke: Analysis of SOS-induced spontaneous prophage induction in *Corynebacterium glutamicum* at the single-cell level. *Journal of Bacteriology*, **196**:1 (2014), pp. 180–8. doi: 10.1128/JB.01018-13. (see pp. 12, 93, 130, 132, 134, 137, 142, 150).

[47] J. W. Little and C. B. Michalowski: Stability and instability in the lysogenic state of phage lambda. *Journal of Bacteriology*, **192**:22 (2010), pp. 6064–6076. doi: 10.1128/JB.00726-10. (see p. 12).

[48] T. Nagai, K. Ibata, E. S. Park, M. Kubota, K. Mikoshiba, and A. Miyawaki: A variant of yellow fluorescent protein with fast and efficient maturation for cell-biological applications. *Nature Biotechnology*, **20**: (2002), pp. 87–90. doi: 10.1038/nbt0102-87 (see p. 13).

[49] R. L. Strack, B. Hein, D. Bhattacharyya, S. W. Hell, R. J. Keenan, and B. S. Glick: A rapidly maturing far-red derivative of DsRed-Express2 for whole-cell labeling. *Biochemistry*, **48**:Table 1 (2009), pp. 8279–8281. doi: 10.1021/bi900870u (see p. 13).

[50] G. Whitesides: The origins and the future of microfluidics. *Nature*, **442**:7101 (2006), pp. 368–73. doi: 10.1038/nature05058. (see p. 14).

[51] D. Wang and S. Bodovitz: Single cell analysis: the new frontier in 'omics'. *Trends in Biotechnology*, **28**:6 (2010), pp. 281–90. doi: 10.1016/j.tibtech.2010.03.002. (see p. 14).

[52] E. K. Sackmann, A. L. Fulton, and D. J. Beebe: The present and future role of microfluidics in biomedical research. *Nature*, **507**:7491 (2014), pp. 181–9. doi: 10.1038/nature13118. (see pp. 14, 83).

[53] F. J. H. Hol and C. Dekker: Zooming in to see the bigger picture: Microfluidic and nanofabrication tools to study bacteria. *Science*, **346**:6208 (2014), pp. 1251821–1251821. doi: 10.1126/science.1251821. (see pp. 14, 83, 97).

[54] C. Dusny and A. Schmid: Microfluidic single cell analysis links boundary environments and individual microbial phenotypes. *Environmental Microbiology*, (2014), n/a–n/a. doi: 10.1111/1462-2920.12667. (see pp. 14, 111).

[55] J. W. Young, J. C. W. Locke, A. Altinok, N. Rosenfeld, T. Bacarian, P. S. Swain, E. Mjolsness, and M. B. Elowitz: Measuring single-cell gene expression dynamics in bacteria using fluorescence time-lapse microscopy. *Nature Protocols*, **7**:1 (2012), pp. 80–8. doi: 10.1038/nprot.2011.432. (see p. 14).

[56] A. Grünberger, S. Helfrich, C. Probst, W. Wiechert, K. Nöh, and D. Kohlheyer: High-Throughput lineage tree investigations of bacteria microcolonies using arrays of monolayer growth chambers. In: *Proceedings of the μTAS 2012 Conference*. Okinawa, Japan, 2012 (see pp. 15, 216).

[57] D. Mark, S. Haeberle, G. Roth, F. von Stetten, and R. Zengerle: Microfluidic lab-on-a-chip platforms: requirements, characteristics and applications. *Chemical Society Reviews*, **39**:3 (2010), pp. 1153–82. doi: 10.1039/b820557b. (see p. 15).

[58] A. E. Vasdekis and G. Stephanopoulos: Review of methods to probe single cell metabolism and bioenergetics. *Metabolic engineering*, (2014). doi: 10.1016/j.ymben.2014.09.007. (see p. 15).

[59] A. Grünberger, C. Probst, A. Heyer, W. Wiechert, J. Frunzke, and D. Kohlheyer: Microfluidic picoliter bioreactor for microbial single-cell analysis: fabrication, system setup, and operation. *Journal of Visualized Experiments*, 82 (2013), p. 50560. doi: 10.3791/50560. (see pp. 16, 35, 106, 167, 182).

[60] P. Wang, L. Robert, J. Pelletier, W. L. Dang, F. Taddei, A. Wright, and S. Jun: Robust growth of *Escherichia coli*. *Current Biology*, **20**:12 (2010), pp. 1099–103. doi: 10.1016/j.cub.2010.04.045. (see pp. 16, 83, 111, 114).

[61] C. Probst, A. Grünberger, S. Helfrich, N. Braun, K. Nöh, W. Wiechert, and D. Kohlheyer: Rapid inoculation of single bacteria into parallel picoliter fermentation chambers. *Analytical Methods*, (2014). doi: 10.1039/C4AY02257B. (see pp. 17, 99, 101, 104, 215).

[62] D. B. Murphy: Fundamentals of Light Microscopy and Electronic Imaging. First edit. Wiley Liss, 2001 (see p. 18).

[63] Nikon ND2 Information. url: https://www.openmicroscopy.org/site/support/bio-formats5.1/formats/nikon-nis-elements-nd2.html (visited on 06/19/2015) (see p. 19).

[64] K. W. Eliceiri, M. R. Berthold, I. G. Goldberg, L. Ibáñez, B. S. Manjunath, M. E. Martone, R. F. Murphy, H. Peng, A. L. Plant, B. Roysam, N. Stuurmann, J. R. Swedlow, P. Tomancak, and A. E. Carpenter: Biological Imaging Software Tools. *Nature Methods*, **9**:7 (2012), pp. 697–710. doi: 10.1038/nmeth.2084. (see p. 23).

[65] F. Amat, W. Lemon, D. P. Mossing, K. McDole, Y. Wan, K. Branson, E. W. Myers, and P. J. Keller: Fast, accurate reconstruction of cell lineages from large-scale fluorescence microscopy data. *Nature Methods*, **11**:9 (2014). doi: 10.1038/nmeth.3036. (see p. 23).

[66] A. C. Bovik: The Essential Guide to Image Processing. 2009 (see p. 24).

[67] A. H. K. Roeder, A. Cunha, M. C. Burl, and E. M. Meyerowitz: A computational image analysis glossary for biologists. *Development (Cambridge, England)*, **139**:17 (2012), pp. 3071–80. doi: 10.1242/dev.076414. (see p. 24).

[68] D. Lowe: Object recognition from local scale-invariant features. In: *Proceedings of the Seventh IEEE International Conference on Computer Vision*. Vol. 2. Ieee, 1999, pp. 1150–1157. doi: 10.1109/ICCV.1999.790410. (see p. 28).

[69] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool: Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, **110**: (2008), pp. 346–359. (see p. 28).

[70] M. Calonder, V. Lepetit, C. Strecha, and P. Fua: BRIEF : Binary Robust Independent Elementary Features. *Proceedings of the European Conference on Computer Vision. ECCV'10*, (2010), pp. 778–792 (see p. 28).

[71] P. Thévenaz, U. E. Ruttimann, and M. Unser: A pyramid approach to subpixel registration based on intensity. *IEEE Transactions on Image Processing*, **7**:1 (1998), pp. 27–41. doi: 10.1109/83.650848. (see pp. 29–31).

[72] B. Zitová and J. Flusser: Image registration methods: a survey. *Image and Vision Computing*, **21**:11 (2003), pp. 977–1000. doi: 10.1016/S0262-8856(03)00137-9. (see pp. 29, 32).

[73] S. Saxena and R. K. Singh: A Survey of Recent and Classical Image Registration Methods. *International Journal of Signal Processing, Image Processing and Pattern Recognition*, **7**:4 (2014), pp. 167–176. doi: 10.14257/ijsip.2014.7.4.16. (see p. 29).

[74] B. D. Lucas and T. Kanade: An Iterative Image Registration Technique with an Application to Stereo Vision. In: *Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI 81)*. 1981, pp. 674–679 (see pp. 29–30).

[75] I. Matthews: Lucas-Kanade 20 Years On : A Unifying Framework : Part 1. *International Journal of Computer Vision*, (2004). (see p. 30).

[76] K. Li: The image stabilizer plugin for ImageJ. 2008. URL: http://www.cs.cmu.edu/~kangli/code/Image_Stabilizer.html (visited on 05/12/2015) (see p. 30).

[77] R. C. Gonzalez and R. E. Woods: Digital Image Processing. 3rd ed. Prentice Hall, 2008 (see pp. 31–32, 44, 46).

[78] D. Ballard: Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition*, **13**:2 (1981), pp. 111–122. DOI: 10.1016/0031-3203(81)90009-1 (see p. 33).

[79] C. Jung and R. Schramm: Rectangle detection based on a windowed Hough transform. *Proceedings of the 17th Brazilian Symposium on Computer Graphics and Image Processing*, (), pp. 113–120. DOI: 10.1109/SIBGRA.2004.1352951. (see p. 33).

[80] R. Brunelli: Template Matching Techniques in Computer Vision: Theory and Practice. John Wiley & Sons, Ltd., 2009, pp. 1–348 (see pp. 33, 35, 41).

[81] J. Canny: A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **PAMI-8**:6 (1986). DOI: 10.1109/TPAMI.1986.4767851 (see p. 34).

[82] M. Sezgin and B. Sankur: Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging*, **13**:1 (2004), pp. 146–165. DOI: 10.1117/1.1631316. (see p. 38).

[83] N. Otsu: A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, **9**:1 (1979), pp. 62–66. DOI: 10.1109/TSMC.1979.4310076. (see pp. 38–39).

[84] J. Sauvola and M. Pietikäinen: Adaptive document image binarization. *Pattern Recognition*, **33**:2 (2000), pp. 225–236. DOI: 10.1016/S0031-3203(99)00055-2. (see pp. 38, 40, 46).

[85] N. Phansalkar and S. More: Adaptive local thresholding for detection of nuclei in diversity stained cytology images. *Proceedings of the International Conference on Communications and Signal Processing (ICCSP11)*, (2011), pp. 218–220. (see p. 40).

[86] J. Bernsen: Dynamic thresholding of gray-level images. In: *International Conference on Pattern Recognition*. 1986 (see p. 40).

[87] J. Roerdink and A. Meijster: The Watershed Transform: Definitions, Algorithms, and Parallelization Strategies. *Fundamenta Informaticae*, **41**: (2000), pp. 1–40. (see p. 41).

[88] L. Vincent and P. Soille: Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**:6 (1991), pp. 583–598. DOI: 10.1109/34.87344. (see p. 41).

[89] M. Kass and A. Witkin: Snakes: Active contour models. *International Journal of Computer Vision*, **1**: (1988), pp. 321–331. (see pp. 42, 52).

[90] H.-Y. Yang, X.-Y. Wang, Q.-Y. Wang, and X.-J. Zhang: LS-SVM based image segmentation using color and texture information. *Journal of Visual Communication and Image Representation*, **23**:7 (2012), pp. 1095–1112. doi: 10.1016/j.jvcir.2012.07.007. (see p. 43).

[91] T. Tran, R. Wehrens, and L. Buydens: Clustering multispectral images: a tutorial. *Chemometrics and Intelligent Laboratory Systems*, **77**: (2005), pp. 3–17. doi: 10.1016/j.chemolab.2004.07.011. (see p. 43).

[92] G. Hamerly and C. Elkan: Alternatives to the k-means algorithm that find better clusterings. In: *Proceedings of the eleventh international conference on Information and knowledge management*. Vol. 4. 09. 2002, pp. 600–607. doi: 10.1145/584887.584890. (see p. 43).

[93] J. W. Robbin and D. Salamon: Dynamical systems, shape theory and the Conley index. *Ergodic Theory and Dynamical Systems*, **8**:1988 (1988), p. 375. doi: 10.1017/S0143385700009494 (see p. 47).

[94] J. J. Koenderink and A. J. van Doorn: Surface shape and curvature scales. *Image and Vision Computing*, **10**:8 (1992), pp. 557–564. doi: 10.1016/0262-8856(92)90076-F (see pp. 47–48).

[95] X. Bai, C. Sun, and F. Zhou: Splitting touching cells based on concave points and ellipse fitting. *Pattern Recognition*, **42**:11 (2009), pp. 2434–2446. doi: 10.1016/j.patcog.2009.04.003. (see p. 50).

[96] Z. Wang, M. B. Salah, H. Zhang, and N. Ray: Shape based appearance model for kernel tracking. *Image and Vision Computing*, **30**:4-5 (2012), pp. 332–44. doi: 10.1016/j.imavis.2012.03.003. (see p. 50).

[97] R. Delgado-Gonzalo, P. Thévenaz, C. S. Seelamantula, and M. Unser: Snakes with an ellipse-reproducing property. *IEEE Transactions on Image Processing*, **21**:3 (2012), pp. 1258–71. doi: 10.1109/TIP.2011.2169975. (see pp. 51–52).

[98] K. Bokelmann: High-Precision Segmentation of Rod-Shaped Bacteria Using Parametric Snakes. Master's thesis. Ernst Moritz Arndt Universität Greifswald, 2014 (see pp. 51, 70, 81).

[99] V. Caselles, R. Kimmel, and G. Sapiro: Geodesic Active Contours. *International Journal of Computer Vision*, **22**:1 (1997), pp. 61–79. doi: 10.1023/A:1007979827043. (see p. 52).

[100] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky: Fast geodesic active contours. *IEEE Transactions on Image Processing*, **10**:10 (2001), pp. 1467–75. doi: 10.1023/A:1007979827043. (see p. 52).

[101] T. F. Chan and L. a. Vese: Active contours without edges. *IEEE Transactions on Image Processing*, **10**:2 (2001), pp. 266–77. doi: 10.1109/83.902291. (see p. 52).

[102] O. Dzyubachyk, W. a. van Cappellen, J. Essers, W. J. Niessen, and E. Meijering: Advanced level-set-based cell tracking in time-lapse fluorescence microscopy. *IEEE Transactions on Medical Imaging*, **29**:3 (2010), pp. 852–67. doi: 10.1109/TMI.2009.2038693. (see pp. 52, 56).

[103] M. Jacob, T. Blu, and M. Unser: Efficient energies and algorithms for parametric snakes. *IEEE Transactions on Image Processing*, **13**:9 (2004), pp. 1231–1244. doi: 10.1109/TIP.2004.832919 (see p. 52).

[104] D. Schmitter, P. Wachowicz, D. Sage, A. Chasapi, I. Xenarios, V. Simanis, and M. Unser: A 2D/3D image analysis system to track fluorescently labeled structures in rod-shaped cells: application to measure spindle pole asymmetry during mitosis. *Cell Division*, **8**:6 (2013), pp. 1–13. (see p. 52).

[105] E. Meijering: Cell Segmentation: 50 Years Down the Road. *IEEE Signal Processing Magazine*, **29**:5 (2012), pp. 140–145. doi: 10.1109/MSP.2012.2204190 (see pp. 54, 83).

[106] G. Myers: Why bioimage informatics matters. *Nature Methods*, **9**:7 (2012), pp. 659–60. doi: 10.1038/nmeth.2024. (see p. 54).

[107] A. Cardona and P. Tomancak: Current challenges in open-source bioimage informatics. *Nature Methods*, **9**:7 (2012), pp. 661–5. doi: 10.1038/nmeth.2082. (see p. 54).

[108] E. Meijering, O. Dzyubachyk, and I. Smal: Methods for cell and particle tracking. *Methods in Enzymology*, **504**: (2012), pp. 183–200. doi: 10.1016/B978-0-12-391857-4.00009-4. (see pp. 55–56).

[109] M. K. Cheezum, W. F. Walker, and W. H. Guilford: Quantitative comparison of algorithms for tracking single fluorescent particles. *Biophysical Journal*, **81**:4 (2001), pp. 2378–88. doi: 10.1016/S0006-3495(01)75884-5. (see p. 56).

[110] W. J. Godinez, M. Lampe, S. Wörz, B. Müller, R. Eils, and K. Rohr: Deterministic and probabilistic approaches for tracking virus particles in time-lapse fluorescence microscopy image sequences. *Medical Image Analysis*, **13**:2 (2009), pp. 325–342. doi: 10.1016/j.media.2008.12.004. (see p. 56).

[111] N. Paragios and R. Deriche: Geodesic active regions and level set methods for motion estimation and tracking. *Computer Vision and Image Understanding*, **97**:3 (2005), pp. 259–282. doi: 10.1016/j.cviu.2003.04.001. (see p. 56).

[112] Particle Tracking Challenge. url: http://www.bioimageanalysis.org/track/ (visited on 06/20/2015) (see p. 56).

[113] M. Maška, V. Ulman, D. Svoboda, P. Matula, P. Matula, C. Ederra, A. Urbiola, T. España, S. Venkatesan, D. M. W. Balak, P. Karas, T. Bolcková, M. Streitová, C. Carthel, S. Coraluppi, N. Harder, K. Rohr, K. E. G. Magnusson, J. Jaldén, H. M. Blau, O. Dzyubachyk, P. Křížek, G. M. Hagen, D. Pastor-Escuredo, D. Jimenez-Carretero, M. J. Ledesma-Carbayo, A. Muñoz-Barrutia, E. Meijering, M. Kozubek, and C. Ortiz-de-Solorzano: A benchmark for comparison of cell tracking algorithms. *Bioinformatics (Oxford, England)*, **30**:11 (2014), pp. 1609–17. doi: 10.1093/bioinformatics/btu080. (see p. 56).

[114] N. Chenouard, I. Smal, F. de Chaumont, M. Maška, I. F. Sbalzarini, Y. Gong, J. Cardinale, C. Carthel, S. Coraluppi, M. Winter, A. R. Cohen, W. J. Godinez, K. Rohr, Y. Kalaidzidis, L. Liang, J. Duncan, H. Shen, Y. Xu, K. E. G. Magnusson, J. Jaldén, H. M. Blau, P. Paul-Gilloteaux, P. Roudot, C. Kervrann, F. Waharte, J.-Y. Tinevez, S. L. Shorte, J. Willemse, K. Celler, G. P. van Wezel, H.-W. Dan, Y.-S. Tsai, C. Ortiz de Solórzano, J.-C. Olivo-Marin, and E. Meijering: Objective comparison of particle tracking methods. *Nature Methods*, **11**:3 (2014), pp. 281–289. doi: 10.1038/nmeth.2808. (see pp. 56–57).

[115] K. Jaqaman, D. Loerke, M. Mettlen, H. Kuwata, S. Grinstein, S. L. Schmid, and G. Danuser: Robust single-particle tracking in live-cell time-lapse sequences. *Nature Methods*, **5**:8 (2008), pp. 695–702. doi: 10.1038/nmeth.1237. (see pp. 57–58, 60–61, 72, 183).

[116] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein: Introduction to algorithms. 3rd ed. Cambridge, MA: MIT Press, 2009, pp. 1–1292 (see p. 59).

[117] H. W. Kuhn: The Hungarian method for the assignment problem. In: *50 Years of Integer Programming 1958-2008: From the Early Years to the State-of-the-Art*. 2010, pp. 29–47. doi: 10.1007/978-3-540-68279-0_2 (see p. 59).

[118] R. Jonker and a. Volgenant: A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing*, **38**:4 (1987), pp. 325–340. doi: 10.1007/BF02278710 (see p. 59).

[119] O. Al-Kofahi, R. J. Radke, S. K. Goderie, Q. Shen, S. Temple, and B. Roysam: Automated cell lineage construction: a rapid method to analyze clonal development established with murine neural progenitor cells. *Cell cycle (Georgetown, Tex.)*, **5**:3 (2006), pp. 327–35. (see pp. 61, 63).

[120] u-track. url: http://lccb.hms.harvard.edu/software.html (visited on 06/20/2015) (see p. 64).

[121] TrackMate. url: http://fiji.sc/TrackMate (visited on 06/20/2015) (see p. 64).

[122] J. Schindelin, I. Arganda-Carreras, E. Frise, V. Kaynig, M. Longair, T. Pietzsch, S. Preibisch, C. Rueden, S. Saalfeld, B. Schmid, J.-Y. Tinevez, D. J. White, V. Hartenstein, K. Eliceiri, P. Tomancak, and A. Cardona: Fiji: an open-source platform for biological-image analysis. *Nature Methods*, **9**:7 (2012), pp. 676–682. doi: 10.1038/nmeth.2019. (see pp. 65, 183).

[123] T. Pietzsch, S. Preibisch, P. Tomancák, and S. Saalfeld: ImgLib2–generic image processing in Java. *Bioinformatics (Oxford, England)*, **28**:22 (2012), pp. 3009–11. doi: 10.1093/bioinformatics/bts543. (see p. 65).

[124] SciJava. url: http://www.scijava.org/ (visited on 06/20/2015) (see p. 65).

[125] C. A. Schneider, W. S. Rasband, and K. W. Eliceiri: NIH Image to ImageJ: 25 years of image analysis. *Nature Methods*, **9**:7 (2012), pp. 671–5. doi: 10.1038/nmeth.2089. (see pp. 66, 182).

[126] M. Linkert, C. T. Rueden, C. Allan, J.-M. Burel, W. Moore, A. Patterson, B. Loranger, J. Moore, C. Neves, D. Macdonald, A. Tarkowska, C. Sticco, E. Hill, M. Rossner, K. W. Eliceiri, and J. R. Swedlow: Metadata matters: access to image data in the real world. *The Journal of Cell Biology*, **189**:5 (2010), pp. 777–82. doi: 10.1083/jcb.201004104. (see p. 66).

[127] StackReg. url: http://bigwww.epfl.ch/thevenaz/stackreg/ (visited on 06/20/2015) (see p. 69).

[128] TurboReg. url: http://bigwww.epfl.ch/thevenaz/turboreg/ (visited on 06/20/2015) (see p. 69).

[129] S. Helfrich, C. E. Azzouzi, C. Probst, J. Seiffarth, A. Grünberger, W. Wiechert, D. Kohlheyer, and K. Nöh: Vizardous: interactive analysis of microbial populations with single cell resolution: Fig. 1. *Bioinformatics*, **31**:23 (2015), pp. 3875–3877. doi: 10.1093/bioinformatics/btv468. (see pp. 83, 183, 215, 217).

[130] J.-B. Nobs and S. J. Maerkl: Long-term single cell analysis of S. pombe on a microfluidic microchemostat array. *PloS one*, **9**:4 (2014), e93466. doi: 10.1371/journal.pone.0093466. (see p. 83).

[131] B. L. Wang, A. Ghaderi, H. Zhou, J. Agresti, D. a. Weitz, G. R. Fink, and G. Stephanopoulos: Microfluidic high-throughput culturing of single cells for selection based on extracellular metabolite production or consumption. *Nature Biotechnology*, **32**:5 (2014), pp. 473–8. doi: 10.1038/nbt.2857. (see p. 83).

[132] S. Taheri-Araghi, S. D. Brown, J. T. Sauls, D. B. McIntosh, and S. Jun: Single-Cell Physiology. *Annual Review of Biophysics*, **44**:1 (2014). doi: 10.1146/annurev-biophys-060414-034236. (see pp. 83, 111, 114).

[133] J. H. Levine, Y. Lin, and M. B. Elowitz: Functional roles of pulsing in genetic circuits. *Science*, **342**:6163 (2013), pp. 1193–1200. doi: 10.1126/science.1239999 (see p. 83).

[134] A. Z. Rosenthal and M. B. Elowitz: Following evolution of bacterial antibiotic resistance in real time. *Nature Genetics*, **44**:1 (2012), pp. 11–3. (see p. 83).

[135] M. V. Han and C. M. Zmasek: phyloXML: XML for evolutionary biology and comparative genomics. *BMC Bioinformatics*, **10**:356 (2009). doi: 10.1186/1471-2105-10-356. (see pp. 84, 86).

[136] Jython. url: http://www.jython.org (visited on 06/20/2015) (see p. 85).

[137] C. Allan, J.-M. Burel, J. Moore, C. Blackburn, M. Linkert, S. Loynton, D. MacDonald, W. J. Moore, C. Neves, A. Patterson, M. Porter, A. Tarkowska, B. Loranger, J. Avondo, I. Lagerstedt, L. Lianas, S. Leo, K. Hands, R. T. Hay, A. Patwardhan, C. Best, G. J. Kleywegt, G. Zanetti, and J. R. Swedlow: OMERO: flexible, model-driven data management for experimental biology. *Nature Methods*, **9**:3 (2012), pp. 245–253. doi: 10.1038/nmeth.1896. (see p. 85).

[138] P. Bille: A Survey on Tree Edit Distance and Related Problems. *Theoretical Computer Science*, **337**:1-3 (2005), pp. 217–239. doi: 10.1016/j.tcs.2004.12.030. (see p. 92).

[139] F. Hufsky, K. Dührkop, F. Rasche, M. Chimani, and S. Böcker: Fast alignment of fragmentation trees. *Bioinformatics (Oxford, England)*, **28**:12 (2012), pp. i265–i273. doi: 10.1093/bioinformatics/bts207. (see p. 92).

[140] D. Binder, A. Grünberger, A. Loeschcke, C. Probst, C. Bier, J. Pietruszka, W. Wiechert, D. Kohlheyer, K.-E. Jaeger, and T. Drepper: Light-responsive control of bacterial gene expression: precise triggering of the *lac* promoter activity using photocaged IPTG. *Integrative Biology*, **6**:8 (2014), pp. 755–765. doi: 10.1039/c4ib00027g. (see pp. 93, 150).

[141] A. Grünberger, C. Probst, S. Helfrich, A. Nanda, B. Stute, W. Wiechert, E. von Lieres, K. Nöh, J. Frunzke, and D. Kohlheyer: Spatiotemporal microbial single-cell analysis using a high-throughput microfluidics cultivation platform. *Cytometry Part A*, **87**:12 (2015), pp. 1101–1115. doi: 10.1002/cyto.a.22779. (see pp. 93, 150, 215).

[142] A. R. Lara, E. Galindo, O. T. Ramírez, and L. a. Palomares: Living with heterogeneities in bioreactors: understanding the effects of environmental gradients on cells. *Molecular Biotechnology*, **34**:3 (2006), pp. 355–381. doi: 10.1385/MB:34:3:355 (see p. 99).

[143] C. Westerwalbesloh, A. Grünberger, B. Stute, S. Weber, W. Wiechert, D. Kohlheyer, and E. V. Lieres: Modeling and CFD simulation of nutrient distribution in picoliter bioreactors for bacterial growth studies on single-cell level. *Lab on a Chip*, **15**: (2015), pp. 4177–4186. doi: 10.1039/c5lc00646e. (see pp. 99–100).

[144] A. Grünberger: Single-Cell Analysis of Microbial Production Strains in Microfluidic Bioreactors. PhD thesis. RWTH Aachen, 2015, pp. 1–176 (see pp. 99, 101, 117).

[145] C. Probst, A. Grünberger, W. Wiechert, and D. Kohlheyer: Polydimethylsiloxane (PDMS) sub-micron traps for single-cell analysis of bacteria. *Micromachines*, **4**:4 (2013), pp. 357–369. doi: 10.3390/mi4040357 (see p. 101).

[146] G. a. Pankey and L. D. Sabath: Clinical relevance of bacteriostatic versus bactericidal mechanisms of action in the treatment of Gram-positive bacterial infections. *Clinical Infectious Diseases*, **38**:6 (2004), pp. 864–870. doi: 10.1086/381972 (see p. 105).

[147] A. M. Feist, D. C. Zielinski, J. D. Orth, J. Schellenberger, M. J. Herrgard, and B. O. Palsson: Model-driven evaluation of the production potential for growth-coupled products of *Escherichia coli*. *Metabolic Engineering*, **12**:3 (2010), pp. 173–186. doi: 10.1016/j.ymben.2009.10.003. (see p. 105).

[148] J. Monod: The growth of bacterial cultures. *Annual Reviews in Microbiology*, Xl (1949). (see p. 106).

[149] M. Peleg and M. G. Corradini: Microbial Growth Curves: What the Models Tell Us and What They Cannot. *Critical Reviews in Food Science and Nutrition*, **51**:10 (2011), pp. 917–45. doi: 10.1080/10408398.2011.570463. (see p. 106).

[150] K. Kaufmann: Fitting and using growth curves. *Oecologia*, (1981), pp. 293–299. (see p. 106).

[151] J. R. Moffitt, J. B. Lee, and P. Cluzel: The single-cell chemostat: an agarose-based, microfluidic device for high-throughput, single-cell studies of bacteria and bacterial communities. *Lab on a Chip*, **12**:8 (2012), p. 1487. doi: 10.1039/c2lc00009a. arXiv: NIHMS150003 (see p. 106).

[152] A. Groisman, C. Lobo, H. Cho, J. K. Campbell, Y. S. Dufour, A. M. Stevens, and A. Levchenko: A microfluidic chemostat for experiments with bacterial and yeast cells. *Nature Methods*, **2**:9 (2005), pp. 685–689. doi: 10.1038/nmeth784 (see p. 106).

[153] H. Kortmann, P. Chasanis, L. M. Blank, J. Franzke, E. Y. Kenig, and A. Schmid: The Envirostat - a new bioreactor concept. *Lab on a Chip*, **9**:4 (2009), pp. 576–585. doi: 10.1039/b809150a (see p. 106).

[154] K. S. Lee, P. Boccazzi, A. J. Sinskey, and R. J. Ram: Microfluidic chemostat and turbidostat with flow rate, oxygen, and temperature control for dynamic continuous culture. *Lab on a Chip*, **11**:10 (2011), pp. 1730–1739. doi: 10.1039/c1lc20019d (see p. 106).

[155] A. E. Vasdekis and G. Stephanopoulos: Review of methods to probe single cell metabolism and bioenergetics. *Metabolic Engineering*, **27**: (2014), pp. 115–135. doi: 10.1016/j.ymben.2014.09.007. (see p. 106).

[156] G. Schwedt: Taschenatlas der Analytik. 3rd ed. Wiley-VCH, 2007, p. 15 (see p. 107).

[157] M. Schaechter, O. MaalOe, and N. O. Kjeldgaard: Dependency on Medium and Temperature of Cell Size and Chemical Composition during Balanced Growth of *Salmonella* typhimurium. *Journal of General Microbiology*, **19**:3 (1958), pp. 592–606. doi: 10.1099/00221287-19-3-592 (see pp. 111, 124).

[158] M. J. Osborn and L. Rothfield: Cell shape determination in *Escherichia coli*. *Current Opinion in Microbiology*, **10**:6 (2007), pp. 606–610. doi: 10.1016/j.mib.2007.09.004 (see p. 111).

[159] M. Osella, E. Nugent, and M. C. Lagomarsino: Concerted control of *Escherichia coli* cell division. *Proceedings of the National Academy of Sciences of the United States of America*, **111**:9 (2014), pp. 3431–3435. doi: 10.1073/pnas.1313715111. (see pp. 111, 126).

[160] M. Campos, I. V. Surovtsev, S. Kato, A. Paintdakhi, B. Beltran, S. E. Ebmeier, and C. Jacobs-Wagner: A Constant Size Extension Drives Bacterial Cell Size Homeostasis. *Cell*, **159**:6 (2014), pp. 1433–1446. doi: 10.1016/j.cell.2014.11.022. (see pp. 114, 125–126).

[161] A. Arndt and B. J. Eikmanns: Regulation of Carbon Metabolism in *Corynebacterium glutamicum*. In: *Corynebacteria: Genomics and Molecular Biology*. Ed. by A. Burkovski. 1st ed. Norfolk: Caister Academic Press, 2008. Chap. 7, pp. 155–182 (see p. 118).

[162] B. Blombach and G. M. Seibold: Carbohydrate metabolism in *Corynebacterium glutamicum* and applications for the metabolic engineering of l-lysine production strains. *Applied Microbiology and Biotechnology*, **86**:5 (2010), pp. 1313–1322. doi: 10.1007/s00253-010-2537-z (see p. 118).

[163] S. Taheri-Araghi, S. Bradde, J. T. Sauls, N. S. Hill, P. A. Levin, J. Paulsson, M. Vergassola, and S. Jun: Cell-Size Control and Homeostasis in Bacteria. *Current Biology*, **25**:3 (2015), pp. 385–391. doi: 10.1016/j.cub.2014.12.009. (see pp. 118, 124–126).

[164] I. Soifer, L. Robert, N. Barkai, and A. Amir: Single-cell analysis of growth in budding yeast and bacteria reveals a common size regulation strategy. **i**: (2014). arXiv: 1410.4771. (see p. 118).

[165] L. Robert: Size sensors in bacteria, cell cycle control, and size control. *Frontiers in Microbiology*, **6**:515 (2015), pp. 1–9. doi: 10.3389/fmicb.2015.00515. (see p. 118).

[166] C. Keilhauer, L. Eggeling, and H. Sahm: Isoleucine synthesis in *Corynebacterium glutamicum*: molecular analysis of the *ilvB-ilvN-ilvC* operon. *Journal of Bacteriology*, **175**:17 (1993), pp. 5595–5603 (see pp. 119, 167, 181).

[167] S. Cooper: Distinguishing between linear and exponential cell growth during the division cycle: single-cell studies, cell-culture studies, and the object of cell-cycle research. *Theoretical Biology & Medical Modelling*, **3**:10 (2006). doi: 10.1186/1742-4682-3-10 (see p. 120).

[168] M. Godin, F. F. Delgado, S. Son, W. H. Grover, A. K. Bryan, A. Tzur, P. Jorgensen, K. Payer, A. D. Grossman, M. W. Kirschner, and S. R. Manalis: Using buoyant mass to measure the growth of single cells. *Nature Methods*, **7**:5 (2010), pp. 387–390. doi: 10.1038/nmeth.1452 (see p. 120).

[169] M. Mir, Z. Wang, Z. Shen, M. Bednarz, R. Bashir, I. Golding, S. G. Prasanth, and G. Popescu: Optical measurement of cycle-dependent cell growth. *Proceedings of the National Academy of Sciences of the United States of America*, **108**:32 (2011), pp. 13124–13129. doi: 10.1073/pnas.1100506108 (see p. 120).

[170] S. Vadia and P. A. Levin: Growth rate and cell size: a re-examination of the growth law. *Current Opinion in Microbiology*, **24**: (2015), pp. 96–103. doi: 10.1016/j.mib.2015.01.011. (see p. 124).

[171] S. Cooper and C. E. Helmstetter: Chromosome replication and the division cycle of *Escherichia coli*. *Journal of Molecular Biology*, **31**:3 (1968), pp. 519–540. doi: 10.1016/0022-2836(68)90425-7 (see p. 125).

[172] W. Donachie: Relationship between cell size and time of initiation of DNA replication. *Nature*, **219**: (1968), pp. 1077–1079. doi: 10.1038/2191077a0. (see p. 125).

[173] A. Amir: Cell size regulation in bacteria. *Physical Review Letters*, **112**:20 (2014), pp. 1–5. doi: 10.1103/PhysRevLett.112.208102 (see pp. 125–126).

[174] A. M. Randich and Y. V. Brun: Molecular mechanisms for the evolution of bacterial morphologies and growth modes. *Frontiers in Microbiology*, **6**: (2015). doi: 10.3389/fmicb.2015.00580. (see p. 127).

[175] T. a. Cameron, J. R. Zupan, and P. C. Zambryski: The essential features and modes of bacterial polar growth. *Trends in Microbiology*, **23**:6 (2015), pp. 347–353. doi: 10.1016/j.tim.2015.01.003. (see p. 127).

[176] C. Donovan, A. Schauss, R. Krämer, and M. Bramkamp: Chromosome segregation impacts on cell growth and division site selection in *Corynebacterium glutamicum*. *PLoS ONE*, **8**:2 (2013), e55078. doi: 10.1371/journal.pone.0055078. (see p. 127).

[177] B. Sieger and M. Bramkamp: Interaction sites of DivIVA and RodA from *Corynebacterium glutamicum*. *Frontiers in Microbiology*, **5**:738 (2015), pp. 1–11. doi: 10.3389/fmicb.2014.00738. (see p. 127).

[178] S. Helfrich, E. Pfeifer, C. Krämer, C. C. Sachs, W. Wiechert, D. Kohlheyer, K. Nöh, and J. Frunzke: Live cell imaging of SOS and prophage dynamics in isogenic bacterial populations. *Molecular Microbiology*, **98**:4 (2015), pp. 636–650. doi: 10.1111/mmi.13147. (see pp. 129, 215).

[179] C. Canchaya, G. Fournous, and H. Brüssow: The impact of prophages on bacterial chromosomes. *Molecular Microbiology*, **53**:1 (2004), pp. 9–18. doi: 10.1111/j.1365-2958.2004.04113.x (see p. 129).

[180] S. Casjens: Prophages and bacterial genomics: What have we learned so far? *Molecular Mirobiology*, **49**:2 (2003), pp. 277–300. doi: 10.1046/j.1365-2958.2003.03580.x (see p. 129).

[181] M. N. Neely and D. I. Friedman: Arrangement and functional identification of genes in the regulatory region of lambdoid phage H-19B, a carrier of a Shiga-like toxin. *Gene*, **223**:1-2 (1998), pp. 105–113. (see p. 129).

[182] M. K. Waldor and J. J. Mekalanos: Lysogenic conversion by a filamentous phage encoding cholera toxin. *Science*, **272**:5270 (1996), pp. 1910–1914. (see p. 129).

[183] J. J. Barondess and J. Beckwith: A bacterial virulence determinant encoded by lysogenic coliphage lambda. *Nature*, **346**:6287 (1990), pp. 871–874. doi: 10.1038/346871a0. (see p. 129).

[184] X. Wang, Y. Kim, Q. Ma, S. H. Hong, K. Pokusaeva, J. M. Sturino, and T. K. Wood: Cryptic prophages help bacteria cope with adverse environments. *Nature Communications*, **1**:9 (2010), p. 147. doi: 10.1038/ncomms1146. (see pp. 129, 140).

[185] G. Edlin, L. Lin, and R. Kudrna: Lambda lysogens of *E. coli* reproduce more rapidly than non-lysogens. *Nature*, **255**:5511 (1975), pp. 735–737. (see p. 129).

[186] A. Lwoff: Lysogeny. *Bacteriological Reviews*, **17**: (1953), pp. 269–332 (see p. 129).

[187] A. M. Nanda, K. M. Thormann, and J. Frunzke: Impact of spontaneous prophage induction on the fitness of bacterial populations and host-microbe interactions. *Journal of Bacteriology*, **197**: (2015), pp. 410–419 (see pp. 129, 140).

[188] M. Carrolo, M. J. Frias, F. R. Pinto, J. Melo-Cristino, and M. Ramirez: Prophage spontaneous activation promotes DNA release enhancing biofilm formation in *Streptococcus pneumoniae*. *PLoS ONE*, **5**:12 (2010), e15678–e15678. doi: 10.1371/journal.pone.0015678 (see pp. 129, 140).

[189] S. A. Rice, C. H. Tan, P. J. Mikkelsen, V. Kung, J. Woo, M. Tay, A. Hauser, D. McDougald, J. S. Webb, and S. Kjelleberg: The biofilm life cycle and virulence of *Pseudomonas aeruginosa* are dependent on a filamentous prophage. *The ISME Journal*, **3**:3 (2009), pp. 271–282. doi: 10.1038/ismej.2008.109. (see pp. 129, 140).

[190] J. Livny and D. I. Friedman: Characterizing spontaneous induction of Stx encoding phages using a selectable reporter system. *Molecular Mirobiology*, **51**:6 (2004), pp. 1691–1704. doi: 10.1046/j.1365-2958.2003.03934.x (see pp. 129, 140).

[191] J. Mitchell, I. R. Siboo, D. Takamatsu, H. F. Chambers, and P. M. Sullam: Mechanism of cell surface expression of the *Streptococcus mitis* platelet binding proteins PblA and PblB. *Molecular Microbiology*, **64**:3 (2007), pp. 844–857. doi: 10.1111/j.1365-2958.2007.05703.x (see pp. 129, 140).

[192] J. R. Fitzgerald, S. R. Monday, T. J. Foster, G. A. Bohach, P. J. Hartigan, W. J. Meaney, and C. J. Smyth: Characterization of a putative pathogenicity island from bovine *Staphylococcus aureus* encoding multiple superantigens. *Journal of Bacteriology*, **183**:1 (2001), pp. 63–70. doi: 10.1128/jb.183.1.63 (see p. 129).

[193] S. Molin and T. Tolker-Nielsen: Gene transfer occurs with enhanced efficiency in biofilms and induces enhanced stabilisation of the biofilm structure. *Current Opinion in Biotechnology*, **14**:3 (2003), pp. 255–261. (see p. 129).

[194] E. C. Friedberg, G. C. Walker, W. Siede, and R. A. Schultz: DNA Repair and Mutagenesis. Washington, 2006 (see pp. 129, 137).

[195] J. D. McCool, E. Long, J. F. Petrosino, H. A. Sandler, S. M. Rosenberg, and S. J. Sandler: Measurement of SOS expression in individual *Escherichia coli* K-12 cells using fluorescence microscopy. *Molecular Microbiology*, **53**:5 (2004), pp. 1343–1357. doi: 10.1111/j.1365-2958.2004.04225.x. (see pp. 130, 137).

[196] J. M. Pennington and S. M. Rosenberg: Spontaneous DNA breakage in single living *Escherichia coli* cells. *Nature Genetics*, **39**:6 (2007), pp. 797–802. doi: 10.1038/ng2051 (see pp. 130, 132, 137, 139).

[197] C. Shee, B. D. Cox, F. Gu, E. M. Luengas, M. C. Joshi, L. Y. Chiu, D. Magnan, J. A. Halliday, R. L. Frisch, J. L. Gibson, R. B. Nehring, H. G. Do, M. Hernandez, L. Li, C. Herman, P. Hastings, D. Bates, R. S. Harris, K. M. Miller, and S. M. Rosenberg: Engineered proteins detect spontaneous DNA breakage in human and bacterial cells. *eLife*, **2**: (2013), e01222. doi: 10.7554/eLife.01222. (see pp. 130, 137, 139).

[198] J. Frunzke, M. Bramkamp, J.-E. Schweitzer, and M. Bott: Population Heterogeneity in *Corynebacterium glutamicum* ATCC 13032 caused by prophage CGP3. *Journal of Bacteriology*, **190**:14 (2008), pp. 5111–9. doi: 10.1128/JB.00310-08. (see pp. 130, 140).

[199] M. M. Cox, M. F. Goodman, K. N. Kreuzer, D. J. Sherratt, S. J. Sandler, and K. J. Marians: The importance of repairing stalled replication forks. *Nature*, **404**:6773 (2000), pp. 37–41. doi: 10.1038/35003501. (see p. 132).

[200] A. Kuzminov: Single-strand interruptions in replicating chromosomes cause double-strand breaks. *Proceedings of the National Academy of Sciences of the United States of America*, **98**:15 (2001), pp. 8241–8246. doi: 10.1073/pnas.131009198. (see pp. 136–137, 139).

[201] S. Pearl, C. Gabay, R. Kishony, A. Oppenheim, and N. Q. Balaban: Nongenetic individuality in the host-phage interaction. *PLoS Biology*, **6**:5 (2008), e120. doi: 10.1371/journal.pbio.0060120. (see pp. 137, 140).

[202] S. Kamensek, Z. Podlesek, O. Gillor, and D. Zgur-Bertok: Genes regulated by the *Escherichia coli* SOS repressor LexA exhibit heterogeneous expression. *BMC Microbiology*, **10**: (2010), p. 283. doi: 10.1186/1471-2180-10-283. (see p. 137).

[203] O. Huisman, R. D'Ari, and S. Gottesman: Cell-division control in *Escherichia coli*: specific induction of the SOS function SfiA protein is sufficient to block septation. *Proceedings of the National Academy of Sciences of the United States of America*, **81**:14 (1984), pp. 4490–4494. (see p. 137).

[204] H. Ogino, H. Teramoto, M. Inui, and H. Yukawa: DivS, a novel SOS-inducible cell-division suppressor in *Corynebacterium glutamicum*. *Molecular Microbiology*, **67**:3 (2008), pp. 597–608. doi: 10.1111/j.1365-2958.2007.06069.x. (see p. 137).

[205] J. W. Drake: Too many mutants with multiple mutations. *Critical Reviews in Biochemistry and Molecular Biology*, **42**:4 (2007), pp. 247–258. doi: 10.1080/10409230701495631. (see p. 137).

[206] J. W. Drake: Mutations in clusters and showers. *Proceedings of the National Academy of Sciences of the United States of America*, **104**:20 (2007), pp. 8203–8204. doi: 10.1073/pnas.0703089104. (see p. 137).

[207] I. Martincorena, A. S. Seshasayee, and N. M. Luscombe: Evidence of non-random mutation rates suggests an evolutionary risk management strategy. *Nature*, **485**:7396 (2012), pp. 95–98. doi: 10.1038/nature10995. (see p. 137).

[208] A. A. Al Mamun, M. J. Lombardo, C. Shee, A. M. Lisewski, C. Gonzalez, D. Lin, R. B. Nehring, C. Saint-Ruf, J. L. Gibson, R. L. Frisch, O. Lichtarge, P. J. Hastings, and S. M. Rosenberg: Identity and function of a large gene network underlying mutagenic repair of DNA breaks. *Science*, **338**:6112 (2012), pp. 1344–1348. doi: 10.1126/science.1226683. (see p. 137).

[209] S. M. Rosenberg, C. Shee, R. L. Frisch, and P. J. Hastings: Stress-induced mutation via DNA breaks in *Escherichia coli*: a molecular mechanism with implications for evolution and medicine. *Bioessays*, **34**:10 (2012), pp. 885–892. doi: 10.1002/bies.201200050. (see p. 137).

[210] R. S. Galhardo, P. J. Hastings, and S. M. Rosenberg: Mutation as a stress response and the regulation of evolvability. *Critical Reviews in Biochemistry and Molecular Biology*, **42**:5 (2007), pp. 399–435. doi: 10.1080/10409230701648502. (see p. 139).

[211] D. Ghosh, K. Roy, K. E. Williamson, S. Srinivasiah, K. E. Wommack, and M. Radosevich: Acyl-homoserine lactones can induce virus production in lysogenic bacteria: an alternative paradigm for prophage induction. *Applied and Environmental Microbiology*, **75**:22 (2009), pp. 7142–7152. doi: 10.1128/AEM.00950-09. (see p. 139).

[212] D. V. Rozanov, R. D'Ari, and S. P. Sineoky: RecA-independent pathways of lambdoid prophage induction in *Escherichia coli*. *Journal of Bacteriology*, **180**:23 (1998), pp. 6306–6315 (see p. 139).

[213] L. Binnenkade, L. Teichmann, and K. M. Thormann: Iron Triggers λSo Prophage Induction and Release of Extracellular DNA in *Shewanella oneidensis* MR-1 Biofilms. *Applied and Environmental Microbiology*, **80**:17 (2014), pp. 5304–5316. doi: 10.1128/AEM.01480-14. (see p. 139).

[214] J. Bondy-Denomy and A. R. Davidson: When a virus is not a parasite: the beneficial effects of prophages on bacterial fitness. *Journal of Microbiology*, **52**:3 (2014), pp. 235–242. doi: 10.1007/s12275-014-4083-3. (see p. 140).

[215] X. Wang, Y. Kim, and T. K. Wood: Control and benefits of CP4-57 prophage excision in *Escherichia coli* biofilms. *The ISME Journal*, **3**:10 (2009), pp. 1164–1179. doi: 10.1038/ismej.2009.59.Control (see p. 140).

[216] J. Gödeke, K. Paul, J. Lassak, and K. M. Thormann: Phage-induced lysis enhances biofilm formation in *Shewanella oneidensis* MR-1. *The ISME Journal*, **5**:4 (2011), pp. 613–626. doi: 10.1038/ismej.2010.153 (see p. 140).

[217] M. E. Zegans, J. C. Wagner, K. C. Cady, D. M. Murphy, J. H. Hammond, and G. a. O'Toole: Interaction between bacteriophage DMS3 and host CRISPR region inhibits group behaviors of *Pseudomonas aeruginosa*. *Journal of Bacteriology*, **191**:1 (2009), pp. 210–219. doi: 10.1128/jb.00797-08 (see p. 140).

[218] R. C. Matos, N. Lapaque, L. Rigottier-Gois, L. Debarbieux, T. Meylheuc, B. Gonzalez-Zorn, F. Repoila, M. D. F. Lopes, and P. Serror: *Enterococcus faecalis* prophage dynamics and contributions to pathogenic traits. *PLoS Genetics*, **9**:6 (2013), e1003539. doi: 10.1371/journal.pgen.1003539. (see p. 140).

[219] N. Q. Balaban, J. Merrin, R. Chait, L. Kowalik, and S. Leibler: Bacterial persistence as a phenotypic switch. *Science (New York, N.Y.)*, **305**:5690 (2004), pp. 1622–5. doi: 10.1126/science.1099390. (see p. 140).

[220] M. Arnoldini, I. A. Vizcarra, R. Peña-Miller, N. Stocker, M. Diard, V. Vogel, R. E. Beardmore, W.-D. Hardt, and M. Ackermann: Bistable Expression of Virulence Genes in *Salmonella* Leads to the Formation of an Antibiotic-Tolerant Subpopulation. *PLoS Biology*, **12**:8 (2014), e1001928. doi: 10.1371/journal.pbio.1001928. (see p. 140).

[221] H. Olofsson, J. Ripa, and N. Jonzén: Bet-hedging as an evolutionary game: the trade-off between egg size and number. *Proceedings of the Royal Society of London B: Biological Sciences*, **276**:1669 (2009), pp. 2963–2969. doi: 10.1098/rspb.2009.0500 (see p. 140).

[222] K. R. Sanft, S. Wu, M. Roh, J. Fu, R. K. Lim, and L. R. Petzold: StochKit2: Software for Discrete Stochastic Simulation of Biochemical Systems With Events. *Bioinformatics (Oxford, England)*, **27**:17 (2011), pp. 2457–8. doi: 10.1093/bioinformatics/btr401. (see p. 141).

[223] B. Michel: After 30 years of study, the bacterial SOS response still surprises us. *PLoS Biology*, **3**:7 (2005), e255. doi: 10.1371/journal.pbio.0030255. (see p. 142).

[224] N. Friedman, S. Vardi, M. Ronen, U. Alon, and J. Stavans: Precise temporal modulation in the response of the SOS DNA repair network in individual bacteria. *PLoS Biology*, **3**:7 (2005), e238. doi: 10.1371/journal.pbio.0030238. (see pp. 142, 145).

[225] N. Ming, W. Si-Yuan, and O. Qi: Modelling the SOS Response by Semi-Stochastic Simulation. *Chinese Physics Letters*, **25**:7 (2008), pp. 2702–2705. doi: 10.1088/0256-307X/25/7/102. (see p. 142).

[226] T. Nishimura, H. Teramoto, M. Inui, and H. Yukawa: Gene expression profiling of *Corynebacterium glutamicum* during Anaerobic nitrate respiration: induction of the SOS response for cell survival. *Journal of Bacteriology*, **193**:6 (2011), pp. 1327–33. doi: 10.1128/JB.01453-10. (see p. 142).

[227] N. Jochmann, A.-K. Kurze, L. F. Czaja, K. Brinkrolf, I. Brune, A. T. Hüser, N. Hansmeier, A. Pühler, I. Borovok, and A. Tauch: Genetic makeup of the *Corynebacterium glutamicum* LexA regulon deduced from comparative transcriptomics and in vitro DNA band shift assays. *Microbiology (Reading, England)*, **155**:Pt 5 (2009), pp. 1459–77. doi: 10.1099/mic.0.025841-0. (see p. 142).

[228] Y. Shimoni, S. Altuvia, H. Margalit, and O. Biham: Stochastic analysis of the SOS response in *Escherichia coli*. *PLoS ONE*, **4**:5 (2009), e5363. doi: 10.1371/journal.pone.0005363. (see p. 142).

[229] S. Liao: Tensor methods for parameter estimation and bifurcation analysis of stochastic reaction networks (2015), pp. 1–26. arXiv: 1406.7825v2 (see p. 145).

[230] S. Coraluppi and C. Carthel: Multi-stage multiple-hypothesis tracking. *ISIF Journal of Advances in Information Fusion*, **6**:1 (2011), pp. 57–68. (see p. 153).

[231] N. Chenouard, I. Bloch, and J. C. Olivo-Marin: Multiple hypothesis tracking for cluttered biological image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **35**:11 (2013), pp. 2736–2750. doi: 10.1109/TPAMI.2013.97 (see p. 153).

[232] imagej-omero: Server- and client-side communication between ImageJ and OMERO. url: https://github.com/imagej/imagej-omero (visited on 06/19/2015) (see p. 154).

[233] J. Sambrook and D. W. Russell: Molecular Cloning: A Laboratory Manual. 3rd ed. Cold Spring Harbor Laboratory Press, 2001 (see p. 181).

[234] D. G. Gibson, G. A. Benders, C. Andrews-Pfannkoch, E. A. Denisova, H. Baden-Tillson, J. Zaveri, T. B. Stockwell, A. Brownley, D. W. Thomas, M. A. Algire, C. Merryman, L. Young, V. N. Noskov, J. I. Glass, J. C. Venter, C. A. Hutchison, and H. O. Smith: Complete chemical synthesis, assembly, and cloning of a *Mycoplasma genitalium* genome. *Science (New York, N.Y.)*, **319**:5867 (2008), pp. 1215–1220. doi: 10.1126/science.1151721 (see p. 181).

[235] A. Niebisch and M. Bott: Molecular analysis of the cytochrome bc_1-aa_3 branch of the *Corynebacterium glutamicum* respiratory chain containing an unusual diheme cytochrome c_1. *Archives of Microbiology*, **175**:4 (2001), pp. 282–294. doi: 10.1007/s002030100262 (see p. 181).

[236] C. C. Sachs, A. Grünberger, S. Helfrich, C. Probst, W. Wiechert, D. Kohlheyer, and K. Nöh: Image-based Single Cell Profiling: High-Throughput Processing of Mother Machine Experiments. *PLoS ONE*, (2016). Submitted (see p. 183).

[237] A. Grünberger, C. Probst, S. Helfrich, W. Wiechert, K. Nöh, and D. Kohlheyer: Industrial Biotechnology meets Microfluidics – Disposable high-throughput single cell analysis device for industrially relevant bacterial strains. In: *Proceedings of the 3rd European Conference on Microfluidics*. Heidelberg, Germany, 2012 (see p. 215).

Publications and Further Work

Peer-reviewed journal articles

N. Mustafi, A. Grünberger, R. Mahr, S. Helfrich, K. Nöh, B. Blombach, D. Kohlheyer, and J. Frunzke: Application of a genetically encoded biosensor for live cell imaging of L-valine production in pyruvate dehydrogenase complex-deficient *Corynebacterium glutamicum* strains. *PLoS ONE*, **9**:1 (2014), e85731. DOI: 10.1371/journal.pone.0085731.

C. Probst, A. Grünberger, S. Helfrich, N. Braun, K. Nöh, W. Wiechert, and D. Kohlheyer: Rapid inoculation of single bacteria into parallel picoliter fermentation chambers. *Analytical Methods*, (2014). DOI: 10.1039/C4AY02257B.

A. Grünberger, C. Probst, S. Helfrich, A. Nanda, B. Stute, W. Wiechert, E. von Lieres, K. Nöh, J. Frunzke, and D. Kohlheyer: Spatiotemporal microbial single-cell analysis using a high-throughput microfluidics cultivation platform. *Cytometry Part A*, **87**:12 (2015), pp. 1101–1115. DOI: 10.1002/cyto.a.22779.

S. Helfrich, C. E. Azzouzi, C. Probst, J. Seiffarth, A. Grünberger, W. Wiechert, D. Kohlheyer, and K. Nöh: Vizardous: interactive analysis of microbial populations with single cell resolution: Fig. 1. *Bioinformatics*, **31**:23 (2015), pp. 3875–3877. DOI: 10.1093/bioinformatics/btv468.

S. Helfrich, E. Pfeifer, C. Krämer, C. C. Sachs, W. Wiechert, D. Kohlheyer, K. Nöh, and J. Frunzke: Live cell imaging of SOS and prophage dynamics in isogenic bacterial populations. *Molecular Microbiology*, **98**:4 (2015), pp. 636–650. DOI: 10.1111/mmi.13147.

Peer-reviewed conference proceedings

A. Grünberger, C. Probst, S. Helfrich, W. Wiechert, K. Nöh, and D. Kohlheyer: Industrial Biotechnology meets Microfluidics – Disposable high-throughput single cell

analysis device for industrially relevant bacterial strains. In: *Proceedings of the 3rd European Conference on Microfluidics*. Heidelberg, Germany, 2012

A. Grünberger, S. Helfrich, C. Probst, W. Wiechert, K. Nöh, and D. Kohlheyer: High-Throughput lineage tree investigations of bacteria microcolonies using arrays of monolayer growth chambers. In: *Proceedings of the μTAS 2012 Conference*. Okinawa, Japan, 2012

Talks

S. Helfrich and K. Nöh: Population Heterogeneity From Image Sequences to Lineage Trees. CECAD Colloquia Series, Köln (Germany), 2012

S. Helfrich: Analysis of microbial observables on single cell level: Challenges of cell Clusters for microscopy-based Image Analysis. 9th European Congress of Chemical Engineering, Den Haag (The Netherlands), 2013

S. Helfrich, A. Grünberger, D. Kohlheyer, W. Wiechert, and K. Nöh: Investigating single-cell growth of microbial cell factories: is counting cells enough? 2nd International Workshop on Image-based Systems Biology, Jena (Germany), 2014

Posters

S. Helfrich, A. Grünberger, D. Kohlheyer, W. Wiechert, and K. Nöh: Semi-Automated Evaluation of Microbial Observables from High-Throughput Time-lapse Microscopy. German Conference on Bioinformatics 2012, Jena (Germany), 2012

S. Helfrich, C. E. Azzouzi, A. Grünberger, W. Wiechert, and K. Nöh: Explorative Analysis of Cell Population Development: Customizable Interactive Lineage Trees. EMBL Heidelberg "New Approaches and Concepts in Microbiology", Heidelberg (Germany), 2013

S. Helfrich, C. E. Azzouzi, C. Krämer, E. Pfeifer, A. Grünberger, N. Mustafi, W. Wiechert, J. Frunzke, D. Kohlheyer, and K. Nöh: Microbial Populations on the Microfluidic Analyst's Couch: Approaching Cellular Heterogeneity. Bioimage Informatics 2014, Leuven (Belgium), 2014

S. Helfrich, C. Krämer, A. Grünberger, C. E. Azzouzi, E. Pfeifer, J. Frunzke, W. Wiechert, D. Kohlheyer, and K. Nöh: Microfluids as platform for the spatiotemporal analysis of bacterial populations at single-cell level. 30th Symposium on "Mechanisms of Gene Regulation", Düsseldorf (Germany), 2014

Software

- **Jü**lich **N**ext **G**eneration **L**ineage **E**xtractor (JuNGLE)
  - not published
- Vizardous
  - Publication: [129]
  - `https://www.github.com/modsim/vizardous`

Band / Volume 116
Structural and electronic investigations on homo- and hetero-organic layers involving CuPc on silver single crystal surfaces
K. M. Schönauer (2015), x, 148 pp
ISBN: 978-3-95806-112-5

Band / Volume 117
First-principles investigation of inelastic magnetic excitations in nanostructures deposited on surfaces
B. J. Schweflinghaus (2016), v, 204 pp
ISBN: 978-3-95806-115-6

Band / Volume 118
Magnetic, structural, and electronic properties of $NiFe_2O_4$ ultrathin films
M. Hoppe (2016), vii, 118 pp
ISBN: 978-3-95806-122-4

Band / Volume 119
First-principle investigation of displacive response in complex solids
D. A. Klüppelberg (2016), xi, 179 pp
ISBN: 978-395806-123-1

Band / Volume 120
Beam Cooling at COSY and HESR - Theory and Simulation -
Part 1 Theory
H. Stockhorst, T. Katayama and R. Maier (2016), v, 192 pp
ISBN: 978-3-95806-127-9

Band / Volume 121
Scanning tunneling microscopy of single-molecule magnets and hybrid-molecular magnets: Two approaches to molecular spintronics
V. Heß (2016), x, 127 pp
ISBN: 978-3-95806-128-6

Band / Volume 122
Bulk and surface sensitive energy-filtered photoemission microscopy using synchrotron radiation for the study of resistive switching memories
M. C. Patt (2016), viii, 247 pp
ISBN: 978-3-95806-130-9

Band / Volume 123
Group IV Epitaxy for Advanced Nano- and Optoelectronic Applications
S. Wirths (2016), vi, 116, XXX pp
ISBN: 978-3-95806-132-3

Band / Volume 124
Strained Silicon-Germanium/Silicon Heterostructure Tunnel FETs for Low Power Applications
S. Blaeser (2016), iv, 91, xvii pp
ISBN: 978-3-95806-135-4

Band / Volume 125
Nanocavity Arrays for Extracellular Recording and Stimulation of Electroactive Cell Systems
A. Czeschik (2016), x, 162 pp
ISBN: 978-3-95806-144-6

Band / Volume 126
Band Structure Engineering in 3D Topological Insulators Investigated by Angle-Resolved Photoemission Spectroscopy
M. Eschbach (2016), VIII, 153 pp
ISBN: 978-3-95806-149-1

Band / Volume 127
Dynamics in colloid and protein systems: Hydrodynamically structured particles, and dispersions with competing attractive and repulsive interactions
J. Riest (2016), ix, 226 pp
ISBN: 978-3-95806-153-8

Band / Volume 128
Self-purifying $La_{2/3}Sr_{1/3}MnO_3$ epitaxial films: Observation of surface precipitation of Mn_3O_4 particles for excess Mn ratios
A. Steffen (2016), 154 pp
ISBN: 978-3-95806-162-0

Band / Volume 129
Strain and electric field mediated manipulation of magnetism in $La_{(1-x)}Sr_xMnO_3/BaTiO_3$ heterostructures
M. Schmitz (2016), VI, 141 pp
ISBN: 978-3-95806-164-4

Band / Volume 130
High-Throughput Live-Cell Imaging for Investigations of Cellular Heterogeneity in *Corynebacterium glutamicum*
S. Helfrich (2016), xvi, 217 pp
ISBN: 978-3-95806-167-5

Schlüsseltechnologien /
Key Technologies
Band / Volume 130
ISBN 978-3-95806-167-5